

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **06103058 A**

(43) Date of publication of application: **15.04.94**

(51) Int. Cl. **G06F 9/06**  
**H04L 9/00**  
**H04L 9/10**  
**H04L 9/12**

(21) Application number: **05113157**

(22) Date of filing: **14.05.93**

(30) Priority: **15.05.92 US 92 883867**  
**15.05.92 US 92 883868**

(71) Applicant: **FISCHER ADDISON M**

(72) Inventor: **FISCHER ADDISON M**

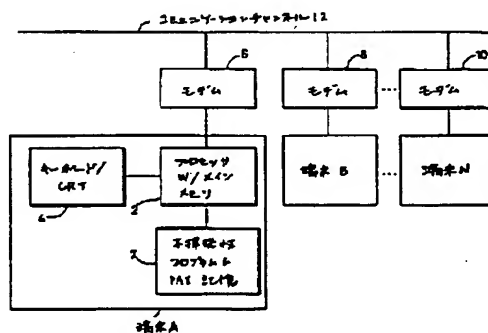
(54) PROGRAM AUTHORITY INFORMATION DATA  
STRUCTURE

(57) Abstract:

PURPOSE: To provide reliable security even in the case of an operation with a complicated data structure, for example, an object including user's own program instructions transmitted between users.

CONSTITUTION: In a computer system provided with a processing means 2 for execution of plural programs and a memory means 7 for storage of program instructions and data, a program authority information data structure includes a means for storage of plural authority entries, and each entry qualifies operations that related programs are allowed to execute, and further, a means which stores data, which relates the authority entry to at least one program, in at least one segment is included.

COPYRIGHT: (C)1994,JPO



(19)日本国特許庁(JP)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平6-103058

(43)公開日 平成6年(1994)4月15日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 9/06

識別記号

4 5 0 C

庁内整理番号

9367-5B

F I

技術表示箇所

H 0 4 L 9/00

9/10

7117-5K

H 0 4 L 9/ 00

Z

審査請求 未請求 請求項の数57(全 28 頁) 最終頁に続く

(21)出願番号 特願平5-113157

(22)出願日 平成5年(1993)5月14日

(31)優先権主張番号 8 8 3 8 6 7

(32)優先日 1992年5月15日

(33)優先権主張国 米国(US)

(31)優先権主張番号 8 8 3 8 6 8

(32)優先日 1992年5月15日

(33)優先権主張国 米国(US)

(71)出願人 593067033

アディソン・エム・フィッシャー

ADDISON M. FISCHER

アメリカ合衆国、33942 フロリダ州、ナ  
ブルズ、フォーティーンズ・アベニュー・サ  
ウス、60

(72)発明者 アディソン・エム・フィッシャー

アメリカ合衆国、33942 フロリダ州、ナ  
ブルズ、フォーティーンズ・アベニュー・サ  
ウス、60

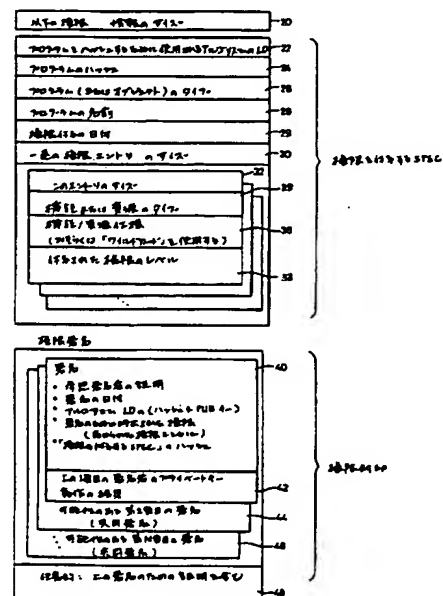
(74)代理人 弁理士 深見 久郎 (外3名)

(54)【発明の名称】 プログラム権限情報データ構造

(57)【要約】

【目的】 ユーザ間で送信されるそれ自身のプログラム命令を含む、複雑なデータ構造、たとえば、オブジェクトを伴って動作する場合でさえ、信頼できるセキュリティを提供する。

【構成】 複数のプログラムを実行するための処理手段(2)と、プログラム命令およびデータを記憶するためのメモリ手段(7)とを有するコンピュータシステムにおいて、プログラム権限情報データ構造は、複数の権限エントリを記憶するための手段を含み、そのエントリは関連のプログラムが実行することを許容される動作を資格付け、少なくとも1つのセグメントにその権限エントリを少なくとも1つのプログラムと関連させるためのデータを記憶するための手段をさらに含む。



## 【特許請求の範囲】

【請求項1】 複数のプログラムを実行するための処理手段(2)と、プログラム命令およびデータを記憶するためのメモリ手段(7)とを有するコンピュータシステムにおいて、プログラム権限情報データ構造(図2)

は、

複数の権限エントリ(32、34、36)を記憶するための手段を含み、前記エントリは関連のプログラムが実行することを許容される動作を資格付け、

少なくとも1つのセグメントに、前記権限エントリを少なくとも1つのプログラム(24、28)と関連させるためのデータを記憶するための手段をさらに含む、データ構造。

【請求項2】 前記少なくとも1つのセグメントは前記関連のプログラム(24)のハッシュを記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項3】 複数の権限エントリを記憶するための手段は前記エントリ(34)の各々について少なくとも1つの型の機能および資源を示すための手段を含む、請求項1に記載のデータ構造。

【請求項4】 複数の権限エントリを記憶するための手段はプログラムに対して認められている権限および能力の資格付けを記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項5】 デジタル署名(40)を記憶するための手段をさらに含む、請求項1に記載のデータ構造。

【請求項6】 署名する当事者に対して認められる権限を示すための手段をさらに含む、請求項5に記載のデータ構造。

【請求項7】 複数のデジタル署名が有効とみなされるべき少なくとも1つの署名について必要であることを示すための手段をさらに含む、請求項5に記載のデータ構造。

【請求項8】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムがアクセスする権限を有するデータの組の指示を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項9】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムがアクセスする権限を有する少なくとも1つのファイルのフィールドの組の指示(212)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項10】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムがプログラムを呼出す権限を有するか否かの指示(210)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項11】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムが電子郵便を生成する権限を有するか否かの指示(222)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項12】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムがデータを他のユーザへ伝送する権限を有するか否かの指示(226)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項13】 複数の権限エントリを記憶するための前記手段は、前記関連のプログラムが文書開放動作を行なう権限を有するか否かの指示(228)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項14】 複数の権限エントリを記憶するための前記手段は、メモリアクセス特権がこのプログラムで権限付与されたという指示(232)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項15】 複数の権限エントリを記憶するための前記手段は、ユーザへ情報を表示する能力に関する前記関連のプログラムの少なくとも1つの修飾の指示(238)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項16】 複数の権限エントリを記憶するための前記手段は、ユーザのために入力請求する能力に関して前記関連のプログラムにおける少なくとも1つの資格付けの指示(247)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項17】 複数の権限エントリを記憶するための前記手段は、ユーザのためにデジタル署名を請求する能力に関して前記関連のプログラムにおける少なくとも1つの資格付けの指示(246)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項18】 複数の権限エントリを記憶するための前記手段は、装置を制御する能力に関して前記関連のプログラムにおける少なくとも1つの資格付けの指示(250)を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項19】 複数の権限エントリを記憶するための前記手段は、アクセスがセキュリティクリアランスによって制限されることの指示を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項20】 記憶するための前記手段は、文書開放動作が行なわれてもよいという指示を記憶するための手段を含む、請求項1に記載のデータ構造。

【請求項21】 前記複数の権限エントリはデジタル署名の一部として含まれる、請求項1に記載のデータ構造。

【請求項22】 複数のプログラムを実行するための処理手段(2)と、少なくとも1つのプログラムを記憶するためのメモリ手段(7)とを含むコンピュータシステムにおいて、データセキュリティのための改良された方法は、

関連のプログラムによって使用されるであろう少なくとも1つのそれらのコンピュータ資源および機能を示す複

数の権限エントリを記憶するためのプログラム権限情報データ構造(図2)を確立するステップと、

前記プログラム権限情報データ構造を格納する(266)ステップと、

プログラム権限情報データ構造を、前記コンピュータシステムによって実行されるべき少なくとも1つのプログラムに関連させ、それによってプログラム権限情報がその関連のプログラムが実行されるとき監視されるために利用可能である(272)ステップとを含む、改良された方法。

【請求項23】 プログラム権限情報データ構造の一部として実行されるべき前記プログラムのデジタルハッシュ(24)を含むステップをさらに含む、請求項22に記載の方法。

【請求項24】 権限エンティティのプライベートキーでプログラム権限情報データの少なくとも一部にデジタル署名するステップをさらに含む、請求項22に記載の方法。

【請求項25】 デジタル署名するステップは、複数のデジタル署名がいかなるデジタル署名も有効であるために要求されるということを示す(44)ことを含む、請求項24に記載の方法。

【請求項26】 デジタル署名するステップは、署名者に認められている権限の少なくとも1つの資格付けを示す(40)ステップを含む、請求項24に記載の方法。

【請求項27】 プログラム権限情報データ構造は関連のプログラムが実行される前にコンピュータユーザによって規定される、請求項22に記載の方法。

【請求項28】 前記コンピュータユーザは特別な特権を有さず、前記ユーザ自身の使用のために前記プログラム権限情報を規定する、請求項27に記載の方法。

【請求項29】 確立するステップは、プログラムがアクセスを有し得る少なくとも1つのデータファイルの少なくとも一部の指示(212)を与える、請求項22に記載の方法。

【請求項30】 確立するステップは、実行されるべきプログラムがアクセスを有する少なくとも1つのファイルの指示を与え、かつ少なくとも1つのファイルに情報を書込む能力を特定する(216)、請求項22に記載の方法。

【請求項31】 確立するステップは、前記プログラムによって呼出される権限を付与されるプログラムの組の指示(220)を与える、請求項22に記載の方法。

【請求項32】 実行されるべきプログラムと関連する前記記憶されたプログラム権限情報を使用して、そうでなければ前記プログラムへ利用可能であろう少なくとも1つの資源および機能の使用を制限するステップをさらに含む、請求項22に記載の方法。

【請求項33】 確立するステップは、情報を伝送する

前記関連のプログラムの権限を支配する少なくとも1つの規則を示す(226)、請求項22に記載の方法。

【請求項34】 少なくとも1つの規則は、情報を伝送する権限が電子郵便を利用するステップを含むことを示す(222)、請求項33に記載の方法。

【請求項35】 確立するステップは、デジタル署名を行なうプログラムの能力を支配する少なくとも1つの規則を与える、請求項22に記載の方法。

【請求項36】 確立するステップは、文書開放動作を行なうプログラムの能力を資格付けする、請求項22に記載の方法。

【請求項37】 確立するステップは、権限のために十分な監視を受けていない機械語命令を実行するプログラムの能力を資格付けする、請求項22に記載の方法。

【請求項38】 確立するステップは、前記関連のプログラムがアクセスすることを許容されるメモリの組を資格付けする(212、216)、請求項22に記載の方法。

【請求項39】 確立するステップは、ユーザへ情報を表示するプログラムの能力を支配する資格付けの組を示す(238)、請求項22に記載の方法。

【請求項40】 確立するステップは、ユーザから情報を請求するプログラムの能力を支配する資格付けの組を示す(242)、請求項22に記載の方法。

【請求項41】 確立するステップは、コンピュータへ結合されるコンピュータ制御された資源を制御するプログラムの能力を支配する資格付けの組を示す(250)、請求項22に記載の方法。

【請求項42】 資格付けの組はモデムを介して情報を伝送するプログラムの能力を支配する、請求項41に記載の方法。

【請求項43】 複数のプログラムを実行するための手段と、実行するための前記手段へ結合され、データおよびプログラム命令を記憶するためのメモリ手段とを有するコンピュータシステムにおいて、コンピュータユーザのために実行するための前記手段によってプログラムを実行するための方法は、

実行されるべきプログラムを識別するステップ(300)と、

プログラム権限情報データ構造がプログラムと関連しているか否かを決定するステップ(304)とを含み、前記プログラム権限データ構造は前記コンピュータユーザに対して利用可能である動作の実行からプログラムの能力を資格付けし、

前記プログラム権限情報データ構造を、前記プログラムと関連しているかを調べるステップ(310、330)と、

関連のプログラムが企図された動作を実行することを許容されるか否か、前記プログラム権限情報の調査から決定するステップ(342)と、

もし前記プログラム権限情報データ構造が、前記プログラムが企図された動作を実行することを許容されないことを示せば、前記動作の性能を抑制するステップ(344、410)とをさらに含む、方法。

【請求項44】 前記プログラムが要求された資源を利用することを許可する前に、前記権限情報をチェックするステップをさらに含む、請求項43に記載の方法。

【請求項45】 前記プログラム権限情報データ構造が前記関連のプログラムで規定される動作の性能を許容するか否かをチェックするステップをさらに含む、請求項43に記載の方法。

【請求項46】 ユーザが予め定められた動作を行なうプログラムを実行する権限を割当てられているか否かを決定するためにチェックを行なうステップ(346)をさらに含む、請求項43に記載の方法。

【請求項47】 前記プログラム権限情報データ構造に関連するいかなるデジタル署名も確認するステップ(316)をさらに含む、請求項43に記載の方法。

【請求項48】 もし前記デジタル署名が有効でなければ、前記プログラムの実行を抑制するためのステップ(322、324)を含む、請求項47に記載の方法。

【請求項49】 前記プログラムの権限情報を前記プログラムを呼出しているルーチンに関連する権限情報と組み合わせるステップをさらに含む、請求項43に記載の方法。

【請求項50】 前記権限情報データ構造は前記プログラムのハッシュを記憶するための手段を含み、前記プログラムのハッシュを計算し、計算されたハッシュを前記記憶されたハッシュと比較するステップをさらに含む、請求項43に記載の方法。

【請求項51】 デジタル署名の署名者と関連する権限を確認するステップ(308)をさらに含む、請求項43に記載の方法。

【請求項52】 前記プログラムの権限情報を、前記プログラムによって呼出されるルーチンと関連する権限情報と組み合わせるステップをさらに含む、請求項43に記載の方法。

【請求項53】 前記権限情報は前記関連のプログラムがアクセスする権限を有するデータの組の指示を含む、請求項43に記載の方法。

【請求項54】 前記プログラム命令は機械語コードで表され、前記動作はプログラム権限データ構造を監視し、補強する制御されたスーパーバイザへの呼出しを介して行なわれる、請求項43に記載の方法。

【請求項55】 前記プログラム権限情報を、自身を別の行先へ伝送するための命令を含む移動プログラムと関連させるステップをさらに含む、請求項43に記載の方法。

【請求項56】 証明を変数として、前記変数が前記プログラムによって演算され得るような移動プログラム内

に記憶するステップをさらに含む、請求項55に記載の方法。

【請求項57】 前記プログラム命令は、前記プログラム権限データ構造を監視し、補強するインタプリタプログラムによって処理される解釈「疑似」コードで表される、請求項43に記載の方法。

【発明の詳細な説明】

【0001】

【発明の分野】この発明は一般にデジタル情報に向上したセキュリティおよび保護を与えるための方法および装置に関する。より特定の、この発明はユーザ間で送信されるコンピュータプログラム、特に出所が不明のプログラムを処理する一方で、向上されたコンピュータシステムセキュリティを与えるための方法および装置に関する。

【0002】

【発明の背景および概要】コンピュータ「ウィルス」の潜在的に破壊的な結果は広く一般に知られてきた。コンピュータウィルスは、実行された場合に、ユーザが予期した動作だけではなく、プログラムに組込まれた予期しない、しばしば破壊的な動作の実行を結果としてもたらすコンピュータプログラムという視点から見られるかもしれない。コンピュータウィルスはまた、実行された場合に、そのコードの一部を取出して、そのようなコードを他のプログラムに置いてそれによって他のプログラムを感染させるプログラムという視点から見られるかもしれない。ウィルスはシステム内の他のプログラムを変更し、システムに様々なトラップを設定し、様々なコントロールプログラムを変え、システムのファイルを消去またはそうでない場合には変更などするかもしれない。

【0003】このようなウィルスはユーザのデータを予期しない方法で損なう、調べるまたは傷をつける望ましくない副作用を有するように典型的に悪意をもって構成される。コンピュータウィルスに伴う問題は、ウィルス制御プログラムがユーザがある必要なデータにアクセスした場合に「暗黙のうちに」典型的に実行され、その結果ユーザは破壊的なプログラムが実行されていることに気づきさえしないという事実によって、しばしばよりひどいものになる。

【0004】この発明はこのようなウィルスからの、およびシステムに基づいて実行するが実際のコンピュータウィルスカリヤではないプログラムからの、保護を提供する。この点に関して、プログラムはコンピュータシステムおよび/または関連データに意図しない逆影響を有し得る。たとえば、実行プログラムはあるユーザデータが第三者に送られることを不注意に起こし得る。このようなプログラムはプログラミングエラーの結果であったかもしれないし、または特定の問題を生じるように故意に設計されたものであるかもしれない。

【0005】先行技術の動作システムはデータをコンピ

ユーザから保護するように典型的に設計される。このようなシステムでは、ユーザはしばしば様々な権限を割当てられ、その後関連権限に基づいてプログラムを実行することが可能である。もしユーザの割当てられた権限を超えるプログラムが実行されていれば、そのようなシステムはそのプログラムの実行を停止させるであろう。このような先行技術のシステムは、コンピュータウイルスなどからコンピュータユーザを適切に保護しない。

【0006】ある「システム」関連ファイルをプログラムによって変更されることから保護するセキュリティシステムがある。しかしながら、このようなシステムはコンピュータユーザをユーザ自身のファイルを実行し変更するプログラムから典型的に保護しない。

【0007】この発明はユーザ間で送信される、それ自身のプログラム命令を含む、複雑なデータ構造、たとえば、オブジェクトを伴って動作する場合でさえ、信頼できるセキュリティを提供することに向けられる。この発明はまたシステムプログラムまたはデータをコンピュータウイルスまたはできの悪いプログラミングの潜在的に壊滅的な結果にさらすことなく、より従来のプログラム、たとえばコンピュータ掲示板からの出所の不確かなものをも処理する場合に、向上したセキュリティを提供する。

【0008】この発明の方法および装置は、まさに実行されようとしているプログラムの能力を、予め規定された資源の使用（たとえばデータファイル、データ書き込み能力など）に制限するシステムモニタを含む独自の動作システム設計を利用する。このシステムモニタはどのプログラムが実行を許可され、および／またはどのプログラムが実行から除外されるかを規定する1組の権限を含むデータ構造を構築する。

【0009】実行されるべきプログラムに割当てられた権限および／または制限の組をここで「プログラム権限情報」（または「PAI」）と呼ぶ。一旦規定されると、プログラム権限情報はその後実行されるべき各プログラムと関連して、それによってプログラムが利用することを許可される資源および機能のタイプを正確に表わす。特定のプログラムに関連するPAIは、コンピュータシステムオーナー／ユーザによって、またはそのコンピュータシステムオーナー／ユーザが絶対的に信用する誰かによって割当てられ得る。

【0010】PAIはプログラムが実行し得る動作の範囲を規定し、および／またはプログラムが実行できない動作を規定する。このプログラムは権限を与えられたものにアクセスすることを許可されるが、それ以外には何もアクセスすることができない。この態様で、プログラムはプログラム能力制限「セーフティボックス」に置かれているものとみなされ得る。この「セーフティボックス」はシステムモニタがプログラムを実行するときはい

つでも、そのプログラムのためのPAIが同様にロードされかつモニタされるように、その後プログラムと関連づけられる。プログラムが機能を実行しようとする、または資源にアクセスしようとする場合には、関連するPAIがその動作が規定されたプログラム制限内であることを確認するためにモニタされる。もしプログラムが権限を付与された制限の範囲を超えて何かを実行しようとするれば、プログラム実行は停止される。

【0011】このように、この発明は実行されるべき任意のプログラムからユーザを有利に保護する。この発明は、プログラムがコンピュータ掲示板または不明の信用性を有する他のユーザのような広い範囲の異なった信頼できない場所から得られる、現在のデータ処理実務に鑑みて特に有利である。

【0012】この発明は上述のPAIがプログラムそれ自体（またはプログラムのハッシュ）とともに、ユーザが信用するあるエンティティによってデジタル署名され得ることを熟慮する。デジタル署名がPAIを有効にするために使用された場合、前述のPAIモニタリングはPAI上のデジタル署名を確認して、それがユーザによって信用されるエンティティに属するものであること、それが適切に権限を付与されていること、およびそれならびにその関連プログラムが手を加えられなかったことを確実にすることを含む。

【0013】この発明は本発明者の米国特許第 4,868,877号および第 5,005,200号に記載されたような階層信用デジタル署名証明システムの使用を熟慮するものであり、これらの特許は引用によりここに援用される。これらの特許の教示に従って、単一のハイレベル権限付与エンティティが安全に権限を委任し、多数の他のエンティティの間でプログラムに権限を付与し、かつ任意のレベルで共同署名を要求することが可能であり、それによって権限付与主体（agent）自身によるエラー、不正の可能性を禁止する。これは単一のソフトウェア認証グループが大口にサービスすることを可能にし、それによって各ユーザに対する1人当たりの費用を大幅に削減する。

【0014】この発明の1つの熟慮された実施例において、プログラムはデータオブジェクトの一部であってもよく、それは高級制御言語で書かれ、この高級言語を実行する標準化されたインタプリタプログラムによって実行される。この場合、インタプリタのタスクの一部は、高級論理で遭遇される機能は実際は許可されることを確認することである。もしこのようなタスクが許可されなければ、インタプリタはそのようなタスクを実行する権限が付与されていないプログラムの実行を抑制する。

【0015】多くの利点がこの発明の使用から生じる。たとえば、この発明は隠されたプログラムまたはウイルスがシステムに導入されることが不可能になるように、プログラムに制限を結びつける役割を有利に果たす。ユーザは、1つの機能のために意図されたプログラムが、

(コンピュータウィルスの広がりをもたらすように)偶然または故意に渡っていき、他の関連のないまたは重要な資源に影響を及ぼすことがないことを確実にするために、実行され得る機能に関する詳細を特定することによって保護される。ここに記載された態様におけるプログラム権限情報の使用によって、ユーザは自分が実行するプログラムから自分自身を保護することが可能である。

【0016】管理主体はプログラムの論理のすべての局面を理解する必要性を伴わずに、プログラムの範囲を効果的に制限することができる。管理人はその意図した機能および定義に基づいてプログラムに権限を付与しかつ制限することが可能であり、それによってプログラム欠陥の危険を軽減させる。この態様で、ソフトウェア「時限爆弾」またはウィルスを植えつけようとするかもしれない乱心したまたはいたずら半分のプログラマーの危険は制限され得る。

【0017】この発明はまたデジタル署名がPAIを確証することを可能にする。このように、プログラムは、すべてのメンバーが共通の高級署名権限を信用している大人口の間で、自由にかつ安全に交換可能である。

【0018】不明の信用性を有するプログラムでさえ、プログラム権限情報が広い範囲の制限に関連づけて、それによって潜在的に有益なプログラムが安全に使用されることを許容した後、たとえ公式の信用証明を持っていなくても、使用することが可能である。

【0019】この発明はまた無限の数の異なった資源および機能が制御されることを可能にする。たとえば、制御され得るいくつかの有用な資源/機能は、プログラムをあるファイルまたはデータセットに制限する能力、電子メールを経てユーザの支配領域の範囲の外にいる誰かにデータを送信する能力、デジタル署名を作るまたは要求するプログラムの能力、あるセキュリティクラスのプログラムへのアクセスを制限する能力などを含む。

【0020】この発明はまたプログラムがデジタル署名動作を実行することができるかどうかを制限し、かつどのようにそのような署名が実行されるべきかを制限する能力を提供する。多くの場合、プログラムがユーザからのデジタル署名を求めることに巻込まれた場合、その署名が与えられようとしているデータをユーザに承知させるかどうかはプログラム次第である。電子データ交換(EDI)トランザクションの場合にそのようである。この場合、いたずら半分のアプリケーションプログラムが、ユーザに1組のデータを見せ、しかも他の組のデータを署名のために送ることは考えられることである。この場合、そのプログラムはユーザを騙してユーザが信じるように仕向けられたものとは完全に異なった情報にデジタル署名させたことになる。この発明はユーザをデジタル署名を求めるプログラムから保護するメカニズムを提供する。

【0021】この発明を使用することによって、一般的

なオブジェクト指向データは、ユーザをウィルスまたはいたずら半分のユーザの潜在的な危険にさらすことなく、ユーザ間で転送され得る。

【0022】この発明のこれらおよび他の特徴は、添付の図面とともに考えられるこの発明の好ましい実施例の以下の説明を読むことによってよりよく理解されるであろう。

【0023】

【実施例の詳細な説明】図1はこの発明とともに使用され得る例証的なコミュニケーションシステムをブロック図形式で示す。このシステムはコミュニケーションチャンネル12を含み、それはたとえばそれを介して端末A、B、...N間のコミュニケーションが行なわれ得る安全にされていないチャンネルであってもよい。コミュニケーションチャンネル12はたとえば電話線であってもよい。端末A、B、...Nは、ほんの一例として、従来のキーボード/CRTディスプレイ4に結合されるプロセッサ(メインメモリを有する)2を有するIBM PCであってもよい。付加的に、各プロセッサは不揮発性プログラムおよびディスクメモリデバイスであり得るプログラム権限情報(PAI)記憶(storage)7に好ましくは結合される。各端末A、B、...Nはまた従来のIBMコミュニケーションボード(図示せず)を含み、それは従来のモデム6、8、10にそれぞれ結合された場合、端末がメッセージを送受信することを可能にする。

【0024】各端末はデジタル署名動作が要求されることであれば何でも実行するメッセージを発生し、かつそのメッセージをコミュニケーションチャンネル12

(またはコミュニケーションチャンネル12に接続され得るコミュニケーションネットワーク(図示せず))に接続される他の端末のいずれにもそのメッセージを送信することが可能である。端末A、B...Nはまた、要求された各メッセージに対して署名確証を実行することが可能である。

【0025】図2は例証的なプログラム権限情報(PAI)データ構造の例示である。PAIは1組の権限付与仕様セグメント22-38、および1組の権限付与署名セグメント40-48(ある状況ではオプションであってもよい)を含む。

【0026】ヘッダセグメント20は権限付与仕様セグメントに先行し、後に続くプログラム権限情報の長さを規定する。フィールド長情報はプログラマーがメモリの関連権限情報の程度を容易に決定することを許容する。このように、もし、たとえばオブジェクト指向データ構造(図5に関連して以下に説明される)が利用可能であれば、フィールド20はプログラム権限情報セグメント116が終了するポイントを識別して、図5に示されるプログラムセグメント118の位置を突き止める役割を果たすであろう。

【0027】セグメント22および24は「ハッシュ」



関連セグメントである。当業者によって理解されるように、「ハッシュ」は「一方向」機能であり、同一の値にハッシュする2つのデータ値を見つけることは計算では実行不可能である。すべての実務的な目的のために、ハッシュ関数をデータのオリジナル集合に適用することから得られた値は、オリジナルデータの偽造不可能な独自の指紋である。もしオリジナルデータが任意の態様で変えられれば、このように変更されたデータのハッシュも同様に異なるであろう。

【0028】関連セグメントのハッシングは、この発明に従って適切に権限を付与されたプログラムが後に手を加えられて、変更されたプログラムを結果としてもたらし可能性がないことを確実にする。セグメント24にプログラムハッシュをストアすることによって、ハッシュは関連プログラムが権限が付与された後に変更されなかったことを保証するために後にチェックされ得る。セグメント22において、特定のハッシングアルゴリズムを独自に識別する識別子がストアされる。

【0029】PAIはプログラム（またはオブジェクト）のタイプを識別するセグメント26を任意に含むことが可能であり、たとえば関連プログラムが機械言語プログラム、特定のタイプの監視プログラム、などであることを示す。プログラムのタイプを識別するデータを与えることによって、このシステムはプログラムによって実行されるべき動作の性質に関する何らかの情報を備える。このような情報は、何か予期しない（そして恐らくはいたずら半分の）ことが発生しているという表示を与えることが可能である。PAIはまたそれが署名されたときのプログラムの名前（セグメント28）および権限付与の日付（セグメント29）を識別するフィールドを含み得る。

【0030】セクション30は以下の一連の権限関連エントリのサイズを規定するセグメントである。このフィールドは残っているエントリが所望されるように区切られることを可能にする。

【0031】後に続く各権限エントリは、特定のエントリのサイズを規定するセグメントを含む（32）。各エントリは同様にそれが関連する機能または資源34のタイプを識別するセグメント34を含む。たとえば、プログラムが他のプログラムにデジタル署名を求める権限を付与する権利を有し得るかどうかなどの、広範囲の機能が規定され得る。セグメント36はセグメント34で識別された包括的タイプ内にある特定の機能／資源を特定する。たとえば、特定のユーザファイルはセグメント36で指定されて、セグメント34で特定された「ファイル」をより特定の識別する。セグメント34および36は、所望されれば、単一セグメントに組合わされ得る。セグメント36の「ワイルドカード」への参照は、たとえば、プログラムが予め定められた接頭辞、または接尾辞を有する任意のファイルにアクセスし得ることを

示すことが意図される。たとえば、指定「A\*」はプログラムは「A」で始まるタグによって識別される任意のファイルにアクセス可能であることを示すであろう。同様に、セグメント36はそのプログラムは「DATA」で終わる任意のファイルにアクセスし得ることを示す、またはそのプログラムは指定された組のファイルにアクセスすることができないことを代替的に示し得る、エントリ\*DATAを含み得る。このようなエントリはまたそのプログラムは任意のプログラムファイルを変えることができることを示し得る。セグメント36はこのようにそのプログラムが何ができるかだけではなく、そのプログラムは何をする権限を付与されていないかを特定し得る。

【0032】図2に示されるセグメント38は与えられた権限のレベルを特定する。たとえば、セグメント38はそのプログラムは予め定められた組のファイルからの読出しを許可する権限のレベルを与えられているが、任意のこのようなファイルを変えるまたは消去する権限は否定されていることを特定し得る。

【0033】もしPAIが異なったユーザに利用可能にされれば（所望の受信者に送信されるプログラムのために）、PAIにとってデジタル署名されることが望ましいかもしれない。単一の組織内においてさえ、オプションの権限署名を含むことは望ましいかもしれない。

【0034】権限署名は署名セグメント40を含む。署名セグメント40は署名者の証明（certificate）への参照、つまり署名者の証明を識別するための識別子を含む。この発明の好ましい実施例に従って、このようなデジタル証明は、ユーザのパブリックキーおよびユーザの名前を含む委任されたエンティティによって作成された（エンティティが満足するほどに正確である）デジタルメッセージ、および恐らくはデジタルメッセージに署名する当事者によってユーザに与えられた権限の表現である。このような署名者の証明は、本発明者の米国特許第 4,868,877号および第 5,005,200号の教示を利用して好ましくは作成され、これらの特許はここにはっきりと引用により援用される。これらの特許に従って、証明は証明者によって与えられつつある権限および課せられている制限ならびにセーフガードを含むように構成され、たとえば、被証明者に対する財政上の制限、および被証明者に与えられる信用のレベルのような、証明者にとっての関心事を示す情報を含む。証明はまた、上に示された米国特許に具体的に教示されるように、被証明者に課せられる共同署名および副署名要求を特定し得る。

【0035】署名セグメント40はまた署名日付、およびハンシユならびにパブリックキーの双方に対するアルゴリズム識別子を含み得る。セグメント40は、たとえばプログラムに権限を付与する権限を与えて、予め定められたファイルを変更するために証明で指定された1つ以上の権限を特定する署名に対して呼出された権限を付



加的に含む。付加的に、この署名はたとえば上述のセグメント20ないし38の全体を含む、権限付与仕様のハッシュを含む。

【0036】セグメント40で識別された項目に関する署名者のプライベートキー動作の結果はセグメント42にストアされる。これはX.500に規定されるような標準的なデジタル署名であってもよいし、または本発明者の上述の米国特許の向上したデジタル署名教示に従うものであってもよい。付加的な(可能な第2ないし可能な第N番目の)署名(共同署名)は、セグメント44、46に示されたようにストアされ得る。任意に、権限署名はまたセグメント48の上の署名に対するデジタル証明を含み得る。代替的に、このような証明は識別されたデータベースからアクセス可能である(が、署名が任意のそのようなデータベースにアクセスする必要性を伴わずに確認され得るように、関連署名のためのデジタル証明を含むことが好ましい)。セグメント40ないし48は上述の権限仕様に関連する権限封印を構成する。ここに参照されるデジタル証明/デジタル署名技術に関するすべてのさらなる詳細は、X.500のような標準的な技術、または上述の米国特許に関連するような向上した技術を含む、任意のデジタル署名技術で実行され得る。

【0037】この発明に従って、PAIは実行されるべきプログラムと関連する。図3ないし図6はプログラム権限情報をプログラムに関連付けるための4つの例証的なアプローチを示す。まず図3を参照して、この図はどのようにプログラム権限情報がアクセス制御下で、プログラムと関連して、ストアされるかを例証する。図3はプログラムのシステムのディレクトリの例証的な概略表現を示す。ディレクトリはプログラム1、2、...N(80、86...92それぞれ)の各々の名前を表すデータを含む。

【0038】各プログラム名前識別子に関連するのは、それぞれインジケータ82、88、94であり、関連プログラム、たとえばプログラム1のディスク98上の場所を識別する(104)。付加的に、プログラム関連識別子の各々に関連するのは、それぞれインジケータ84、90、...96であり、その関連プログラム権限情報、たとえばPAI 1の場所を識別する。プログラム権限情報PAI 1は別個のメモリデバイス100にストアされているとして示されるが、それは所望されればその関連プログラムと同一のメモリ媒体にストアされてもよい。上に示したように、プログラムに関連するプログラム権限情報は、プログラム権限情報がユーザ自身によって発生され、(この場合は署名する必要はない)か、または第三者によって発生され、この場合はPAIはしばしば署名されなければならないかに依存して、デジタル署名されてもよいし署名されなくてもよい。

【0039】図4はPAIをプログラムに関連づける他

のアプローチを示す。このアプローチにおいて、プログラム権限情報110はプログラム112にはめ込まれる。図3に関連して上に説明したように、権限付与情報はPAIの源に依存して、任意にデジタル署名されてもよい。

【0040】図5は、PAIデータ構造がこの発明の一実施例に従うプログラムと関連する重要なアプリケーションを示す。図5は安全な交換可能な「オブジェクト」のための例示的なデータ構造を示す。このデータ構造は委託された権限によって署名され得る。このようなデータ構造の署名により、オブジェクトはユーザからユーザへ安全に送信されることが可能になる。図3、図4、図5および図6に示されるデータ構造は一般的なフォーマットで説明されているが、1992年4月6日に出願され、「移動プログラムを作成し支持しかつ処理するための方法および装置(Method and Apparatus for Creating, Supporting and Processing a Travelling Program)」(米国連続番号07/863,552、Atty. Dkt. 264-30)と題された、本発明者の同時係属中の出願で説明されたような構造であってもよく、この出願は引用によりここに明示的に援用される。

【0041】データ構造は、ほんの一例として、たとえば買注文関連オブジェクトまたは任意の他のタイプの電子デジタルオブジェクトに続くタイプのオブジェクトを規定し得るヘッダセグメント114を含む。プログラム権限情報は、以下により詳細に説明される態様で、オブジェクトの単数または複数のプログラムのための権限を特定するセグメント116にはめ込まれる。

【0042】データ構造はオブジェクトプログラムセグメント118を含み、それはたとえば、関連する買注文がプログラムユーザによって対話的に完了される変数フィールドのためのブランクを残すようにディスプレイされる態様を制御し得る。オブジェクトプログラムはそのようなデータをストアし、出願人の上述の同時係属出願に詳細に説明される態様でそれ自体のコピーを添付データとともに送る。図5に示されるように、このプログラムは数個の論理セグメントに分割されて、オブジェクトの異なった用途を収容し得る。たとえば、このプログラムはデジタル買注文を作った人に、後続の受信者にディスプレイするのとは異なったディスプレイを示し得る。プログラムがプログラムによって指定された受信者によって受信された場合、その受信者は送信されたプログラムの写しを呼出して、たとえば受信者の必要に合わせて作られた買注文のディスプレイを制御する。受信者はすべての受信されたデータを検証し、新しいデータを加えることが可能であり、プログラムはそれ自体を受信者の電子メールシステムによって、たとえば購入された品物を実際に出荷するユーザに送ることが可能である。図5に示されたデータ構造は付加的にオブジェクトに関連するデータセグメント120を含み、それは好ましく

は上述の特許出願に説明されるような「変数」セグメントおよびデータファイルセグメントを含む。データセグメント120はオブジェクトの各バージョンまたはインスタンスに関連するデータが別個にストアされ、別個にアクセス可能であるように区切られてもよい、なぜなら異なったユーザは図5に示されるデータ構造に対して異なった用途を有するかもしれないからである。したがって、データはそれが各ユーザからどのように集められるかに依存して変化することになる。しかしながら、プログラム118は好ましくは各ユーザに対してそのままである。委託された権限はプログラム権限情報(PAI)とともにプログラムに署名する、なぜならプログラムの各実行に回答して入力されるデータよりはむしろ、プログラムそれ自体が権限付与される必要があるからである(データは各実行経路の間変化することが可能であるからであり、かつまた正確なデジタル署名が適切に入力データ上で集められることを確実にすることはプログラムの責任であるからである)。

【0043】図6は多くのユーザが同一のプログラム(イメージ)にアクセスする状況を例証し、各々はそれに関連し、ユーザに所属する特定のファイルに維持される自分自身の(おそらくは別個の)プログラム権限情報129を有する。図6はシステムプログラムディレクトリ131を示し、それはプログラム名前に関連するインジケータによって、プログラムXのディスク132上の場所を識別する。この場合、プログラムXがユーザによって呼出された場合はいつでも、システムはユーザがそのプログラムに関連し得るプライベートなPAI仕様(たとえば133、135、137)を有するかどうかを決定するためにチェックする。このように、異なったユーザは自分自身の必要性および信用の知覚に従ってプログラムを制限し得る。これはたとえば大きな固有の権限を有するユーザ、または非常に重要な情報へのアクセスを与えられたユーザが、ありきたりの目的のための「平凡な」プログラムをしばしば実行しなければならない場合に有用であり得る。この場合、そのような重要なユーザにとって、いくつかの(または多くの、またはすべての)「平凡な」プログラムのまわりに「セーフティボックス」を規定して、その結果そのようなプログラムが自分自身の特に重要なデータに影響を及ぼし得る「トロイの木馬」または他の欠点を不注意に含まないようにすることが分別のあることかもしれない。

【0044】したがって、このようなユーザは一般的なPAI「関連付け」を規定し、その結果保護しているPAIは、重要なデータを処理する選ばれて委託されたいくつかのプログラム(select trusted few programs)をおそらく除いて、すべてのプログラムに自動的に関連し得る。

【0045】この発明はPAI情報が任意の適切な態様で関連づけられることを許容し、その結果原則的にユー

ザはおそらくはより総括的なPAI、またはこのプログラムの製造者によって署名されかつ供給されたPAIとともに組合わされる1つ以上のレベルのPAIを規定することが可能である。

【0046】この発明はプログラムとそのPAIとの間の関連付けが、もし必要であれば、1つのプログラムが複数のPAIと関連することが可能であるように、または逆に1つのPAIが複数のプログラムに適用されることが可能であるように、またはこれらのアプローチの何らかの組合わせが可能であるように非常に一般的に構成され得ることを熟慮する。したがって、単純にするために、本明細書では単一のPAIを単一のプログラムと関連して一般に論じているが、これはいかなる意味においても制限であると考えられてはならないことが理解されなければならない。

【0047】図7は特に不明の信用性のプログラムを実行する場合に、どのようにユーザがプログラム権限情報の使用によって利益を得るかを例示するフローチャートである。ブロック121に示されるように、ユーザは自分がプログラムの作成者について何も知らない興味あるプログラムを実行したいという望みを持つかもしれない。このように、そのプログラムは不明の信用性を有し、たとえば電子掲示板を経てアクセスされたものかもしれないし、テレコミュニケーションチャンネルまたはディスクットを経てユーザの端末に到着したものかもしれない。単なるゲームであると称するこのようなプログラムは、それがウィルスに感染しているかもしれないという重要な危険をはらんでいる。

【0048】ブロック122で示されるように、ユーザはそのプログラムを重要でないまたは消耗ファイルのみに制限するプログラム権限情報を規定することによって保護され得る。所望されれば、ユーザはとにかく任意のファイルを変更することからこのようなプログラムを制限することが可能である。たとえば、ユーザはプログラムにディスプレイスクリーンにイメージをディスプレイし、ゲームをすることに関連する機能を実行することのみを許可し得る。代替的に、もしプログラムが単一のワークファイルを有することがわかれば、PAIデータはそのような単一ファイルの使用のみを許可することが可能である。単一ワークファイルのみにアクセスを制限することによって、不明の信用性のプログラムは他のユーザのプログラムにウィルスを導入することができないし、または他の態様ではシステムプログラム誤動作を開始することができない。このように、この発明に従って、ユーザは、システムプログラムによって、プログラムの任意の特権機能を使用する能力をたとえば完全に排除するようなプログラムによる危険にユーザのシステムがどれくらいさらされるかを決定する。ユーザはそれからたとえばオペレータを促すメニュー駆動型システムによって、PAIをそのシステムに基づいて実行されるべ

きすべてのプログラムに関連づける（またはこのような P A I または P A I の欠如を予め定められた省略時 (default) メカニズムによって関連づける）。システムユーティリティプログラムは、図 9 ないし図 13 と関連して以下に詳細に説明される態様で、プログラム権限情報を作成するために好ましくは使用される。

【0049】 P A I が割当てられた後、そのシステムが関連するプログラムを実行するときはいつでも、システムソフトウェアは（以下に説明される態様において）プログラムが P A I と一貫する態様で安全に実行していることを確実にする。このように、プログラムは「セーフティボックス」に有効に置かれたことになる（124）。

【0050】 図 1 に戻って、不明の信用のプログラムはコミュニケーションチャンネル 12 によって、または端末 A にロードされたフロッピーディスクからシステムに導入され得る。このプログラムはたとえばユーザのプログラムディスクメモリ 7 に初めストアされ得る。その後、キーボード 4 上のユーザは（図 7 のブロック 122 に関して）上で識別されたシステムのプログラムとの対話を介して、そのプログラムがユーザのシステムに基づいて安全に実行できるように、またはおそらくは P A I がプログラムとともに到着し、その場合にはおそらく署名されるように、（図 3 ないし図 6 に示されたような態様で）、プログラムにプログラム権限情報を関連づける。

【0051】 図 8 はこの発明の例証的な実施例に従うプログラムコントロールブロック（P C B）データ構造の例示である。プログラムコントロールブロック 140 はシステムモニタによって使用されて関連するプログラムの実行を制御するデータ構造である。

【0052】 プログラムコントロールブロック 140 には、関連するプログラムがプログラムがその割当てられた権限に従って機能を実行し、資源にアクセスすることを確実にするように実行されているときに、P A I が容易に参照され得るように、プログラム権限情報がロードされる。実行されるべきプログラムと関連するプログラムコントロールブロックは、プログラムによって変更することができない記憶エリアに置かれる。

【0053】 図 8 に示されるように、親プログラム（その P C B は 180 で識別される）はプログラム（P C B 170 を有する）を呼出し、それはひいては図 8 に詳細に示されるプログラム 140 を呼出す。各新しい P C B は「以前の」または呼出しているプログラムコントロールブロックを指す 150 のようなフィールドを含む。フィールドはまた「次の」プログラムコントロールブロックファイルを識別するために使用され得る。

【0054】 呼出されたプログラムが実行を終えた場合、システムはその関連する P C B を実行されたスタックの一番上から取除き、関連するプログラムを記憶から

取除き、関連する権限付与情報を取除き、スタックのその真下のプログラムコントロールブロックにアクセスする。他のプログラムが呼出された場合、スタックの一番上に置かれた新しい P C B が作られるように逆のプロセスが発生し、それはフィールド 150 に示されるように以前の P C B を再び指す。

【0055】 プログラムコントロールブロックはまた、図 8 に示されるメモリセグメント 153 によってたとえば示されるように、関連するプログラムがロードされている記憶の場所に対するポインタであるフィールド 152 を含む。付加的に、プログラムのサイズはフィールド 154 で示される（それはこのようにしてプログラムが実行を終了した場合に解放される記憶の量を示す）。

【0056】 プログラムコントロールブロックのフィールド 156 は、1 つ以上の P A I の記憶（157）の場所を識別する（それは関連するプログラムによって変えることができない記憶のエリアに置かれる）。フィールド 156 によって指された P A I は好ましくは上述の図 2 に示された態様で構成される。

【0057】 フィールド 158 は関連するプログラムのためのエントリアドレスを識別する。もしプログラムがその実行中に他のプログラムを呼出せば、フィールド 158 は、呼出されたプログラムがその実行を完了した後、プログラム実行が再開されるであろうアドレスをストアするために使用される。

【0058】 プログラムコントロールブロックはまた、たとえばプログラム状態ワード（P S W）スタック情報などのような状態情報をストアするための 1 組の場所（160）を含む。プログラムコントロールブロックは、付加的に、もしエラーがプログラムの実行中に発生すれば、エラーまたは終結メッセージに関連する情報をストアするためのフィールド 162 を含む。このようなフィールドは、たとえば、なぜプログラムが不成功に終結したかを識別するための呼出しプログラムに利用可能である。フィールド 162 はプログラムが成功のうちに終結したという表示をストアすることが可能である。

【0059】 プログラムコントロールブロック 140 は、プログラムが終わった場合に（164）漂遊（stray）資源が解放され得るように維持される様々なポインタを付加的に含む。このようなポインタはたとえばプログラマーが解放するのを忘れた資源の解放を許容するために有用である。

【0060】 図 9 ないし図 12 はプログラム権限情報を確立するためのユーティリティプログラムの動作の例証的なシーケンスを例示するフローチャートである。このようなユーティリティプログラムはユーザ、つまりエンドユーザ、エンドユーザの主体または製造者でさえ促して、ユーザのシステムによって実行されるべきプログラムに関連する権限の範囲を規定する。

【0061】 図 9 に示されるように、P A I を確立する

10

20

30

40

50

ためにユーティリティプログラムを入力した後(200)、ユーザはそれに対してP A Iが確立されるべきプログラムの名前を供給するように促される(202)。その後、ユーザはP A Iは署名されるべきか署名されるべきでないかを決定するように促される。P A Iは、もしそれがユーザ自身の使用および保護のためのものであれば、またはもしこのP A Iが満足なアクセスコントロールの下でストアされ得るのであれば、必ずしも署名される必要はない。ブロック204のユーザの入力に依存して、ユーザが署名したいのかまたは署名したくないのかについての決定が行なわれる(206)。もしユーザが署名したいのであれば、ブロック208に示されるように、ユーザの証明が検索され、後のテストングのためにフラグが設定され署名動作が実行されていることを示す。ユーザの証明は従来のデジタル証明であってもよいし、本発明者の米国特許第4,868,877号および第5,005,200号に従う権限の委譲を与える向上されたデジタル証明であってもよい。

【0062】ブロック210で示されるように、ユーザはそれからどの権限がプログラムに割当てられるべきか指定するように促される。後に続く権限(そしてそれらが示される順序)は例示の目的のみのために与えられ、かつこの発明に従って割当てられ得るすべての可能な権限の完全なリストであることは意図されないことが理解されなければならない。

【0063】図9に例示されるように、ファイルアクセス権限が呼出されるべきであるかどうかを決定するためにチェックが行なわれる(212)。ファイルアクセス権限(および以下に参照される他の権限の各々)の選択を与えるために、メニューがユーザに表示されてもよい。ファイルアクセス権限は任意の組のファイルのファイルエレメントもしくはフィールド、任意の組のデータもしくはデータエレメント、または任意の組のファイルなどに関する権限を示すために使用され得ることが理解されなければならない。もしユーザがファイルアクセス権限を選択すれば、ユーザはファイル名またはファイルシステムまたは「ワイルドカード」ファイル名パターンを特定するように促されるであろう(214)。上に説明されたように、たとえば、ワイルドカードファイル名パターンはフォームDATA\*から選択されることが可能であり、その結果プログラムは接頭辞「DATA」で始まる任意のファイル名にアクセスする権限を与えられる。

【0064】その後、ユーザはファイルアクセスのタイプを特定するように促される(216)。この点に関して、ユーザはプログラムの権限を、ファイルから読出す、情報をファイルに挿入する、ファイルの情報を更新する、ファイルから情報を消去する、ファイルを消去する、ファイルを転送するなどだけの中から、1つ以上に制限することを特定し得る。もしファイルアクセスまた

は図9ないし図11で以下に示される他の権限が選択されれば、この選択の表示がストアされ、ルーチンは以下に説明される図12および図13のブロック274に分岐する。

【0065】もしユーザがファイルアクセスを選択しなければ、これはこのプログラムに他のプログラムを呼出す権限を付与するための要求であるかどうかを決定するためにチェックが行なわれる(218)。もしそうであれば、プログラムがそれに基づいて呼出され得る、もしあれば、何の制限または資格が確立されるべきであるかを確認するために決定が行なわれる(221)。そのような資格が規定され組合わされ得る多くの方法がある。たとえば、1つの特定のプログラム名のみが呼出されることが許容されることであってもよいし、またはおそらくある(「ワイルドカード」)パターンに一致する名前を有するプログラムのみが呼出されてもよい。おそらくその基準はライブラリ、または組のライブラリの仕様を含み、そこには許可されるプログラムが存在し得る。

【0066】このプログラムによって呼出されるのに相応しいプログラムを資格付ける他の方法は、呼出されたプログラムが呼出しているプログラムほど大きな権限を有してはならないことを特定することである。代替的に、権限および必要性に依存して(およびどのようにシステムが呼出しているおよび呼出されたプログラムの権限を組合わせるように選択するかに依存して)、呼出されたプログラムが呼出しているプログラム以上の権限を有することを要求することが適切であり得る。実際、この「呼出し権限」資格の一部として、それによって権限が呼出されたプログラムと組合わされることが可能な(たとえば呼出されたプログラムの自然の権限を使用することによって、呼出されたものおよび呼出すものの最も制限的な権限を使用することによって、など)方法を特定することさえ適切であるかもしれない。

【0067】ここで使用されるように、特定された権限の資格または制約または制限または許可へのいずれの参照も、適切な基準の任意の収集に基づいて設定された全体の規則仕様を含むことが意図される。「規則」「組の」、「資格」などの言葉はその最も一般的な意味で使用され、それによって仕様は任意のタイプの規則または規則の複合組によって決定されることが可能であり、それはたとえば、制限なしに、直接仕様による、間接仕様による、排他による、リストによる、「ワイルドカード」規則による、または任意の適切な属性、方法または基準によってエレメントを区別する他の任意の方法によることを含む、任意の属性によってエレメントを区別することが可能である。そのような区別は単一のエレメントのみを含む、すべてのエレメントを除外する、またはすべてのエレメントを含む仕様を包含することが意図される。P A Iは全体または一部において、任意の数の隣接するまたは隣接しないデータのセグメントからなり得

る。適切な文脈において、その文脈のために公式化される予め定められた規則があってもよく、それは任意の明示的な資格がない場合に推定される。

【0068】「示す」、「～を指す」、「のアドレス」などの言葉は、たとえば直接仕様、任意のタイプのポインタ、参照、関連、ハッシュ、結合値共通識別子、などを、制限なく、含む適切な関連の任意のタイプを伝えることが一般に意図され、それは任意のレベルの間接性(indirection)を含んでもよく、明示性であってもよく、または任意の明示の関連がないところでも文脈に適切であれば暗示であってもよい。

【0069】「制限」という言葉は制限の一般的な概念を示すことが意図され、しばしば通常的能力に対する「制約」と共通の意味で使用されるが、制限が通常的能力を超えて規定される状況を反映することもまた意図される。

【0070】この発明は、プログラムの能力を通常ユーザに許された資源にアクセスすることに制限する機能を規定することに主に中心を置くが、適切な環境において、ユーザに通常許容された能力を超えるものに拡張するためにも使用され得る。このように、たとえば、そのP A Iがスーパーバイザによって認められた権限によって署名されるプログラムは、拡張された機能を実行することが許容される。

【0071】いくつかの例証的な規則がP A Iがどのように確認されるべきかに関して与えられたが、特定の実施例は広く変わり得る。示されたように、いくつかの場合、P A Iはまったく署名される必要はなく、それはたとえばユーザ自身がP A Iを規定する場合、または信用された管理者がP A Iを信用されたアクセス制御された記憶にストアする場合などである。P A Iが署名された場合、署名確認が、たとえば本発明者の他の特許、米国特許第 4,868,877号および第 5,005,200号に従って行なわれる任意の数の方法がある。ユーザは最終的なパブリックキー、またはその署名をユーザが信用する証明を規定する以前にストアされた情報を有するようである。

【0072】図9に戻って、もしユーザがプログラム呼出し許可を選択しなければ、これはこのプログラムが呼出され得る状況を特定するための要求であるかどうかを決定するためにチェックが行なわれる(220)。もしそうであれば、そのような権限の制限または資格を確認するための決定が行なわれる(223)。1つのそのような仕様は、プログラムはユーザによって直接呼出されなければならない(かつおそらくこれは任意の仕様の代わりに暗黙値であろう)ことであり、おそらくはこのプログラムは特定のリスト、または上述の「呼出し権限」に類似の特定ライブラリからの名前前でプログラムによって呼出されるだけである。おそらくプログラムはより大きな権限を有する、またはより小さな権限を有するプログラムによって呼出されるだけである。どの規則が

適切であるかは、基礎になるシステムが他のプログラムによって呼出されたプログラムのためのP A I権限をどのように組合わせるかに関連し得る。この資格の他の局面は、このプログラムの権限がどのように呼出しているプログラムの権限と組合わされ得るか、たとえばこのプログラムの有効な権限は呼出しているものによって制限されるかどうかを特定することであり得る。他の多くの可能性もまた利用可能であり、権限の各タイプごとにおそらく異なる。

10 【0073】図10に移って、もしブロック220で識別された権限が選択されなかったならば、プログラムが電子郵便を生成することを許容されるべきか否かを決定するためのチェックが行なわれる(222)。もしそうであれば電子郵便を生成するこの能力が資格付けされるべきか、たとえば或る個人に限定されるべきか否かのチェックが行なわれる。もしそうであれば、このようなならなる資格がユーザによって特定される(224)。

20 【0074】もしブロック222で識別された権限が選択されなければ、ユーザはプログラムが他のユーザヘデータを伝送することを許容されるべきか否かについて問われる(226)。もしそうであれば、ブロック224の上述の処理が行なわれ、この権限に対するいかなる資格も決定する。

30 【0075】もしブロック226で識別された権限が選択されなければ、ブロック228に示されるように、プログラムが「文書解放(document release)」動作を行なうことを許容されるか否かを決定するためのチェックが行なわれる。もしそうであれば、たとえば権限が適用される文書のクラス(たとえば極秘(top secret)、機密(secret)、機密扱い(sensitive)、など)をユーザから決定することによってこの権限に対する資格が選択され、記憶されてもよい。代替的に、解放されるべき文書はセキュリティの観点から「解放」を要求しなくてもよいが、むしろ文書の技術的解放に関連してもよい。いずれにおいても、どのような選択された資格も記録される。

40 【0076】もしブロック228で識別された権限が選択されなければ、プログラムが機械語プログラムを実行することが許容されるべきかを決定するためのチェックが行なわれる(232)。この権限は或るルーチンが機械語プログラムとして不適当に実行するか、または実行されることを防止するのに有用であろう。ユーザは任意の適当な資格を特定するように促されてもよい(233)。もしブロック232で識別された権限が選択されなければ、プログラムが任意の特別なメモリアクセス特権、たとえば或るオペレーティングシステムプログラムのために予約された記憶域へのアクセスが与えられるべきであるか否かを決定するチェックが行なわれる(234)。もしそうであれば、ユーザはこのようなアクセス特権へのいかなる資格も適当なものとして特定すること



が促されるであろう。

【0077】もしブロック234で識別された権限が選択されなければ、プログラムがユーザへ情報を表示する権限を有すべきか否かを決定するチェックが行なわれる(238)。この点で、或るプログラムが或る計算を行なう目的のみを意図されてもよい。このようなプログラムはいかなるユーザ対話も存在してはならないように設計され得る。もしこのようなプログラムが許可なく変更されたとすれば、ユーザへ誤ったメッセージを作成する命令が挿入されているかもしれない。たとえば、システム障害があるということ、および動作を再開するためにユーザが自分の秘密のパスワードを入れることが必要であるということを表示し得る。このようなプログラムは当事者へパスワードを自動的に伝送し得、するとその当事者はパスワードおよびそのような画面上に入れられた他のいかなる情報へもアクセスを有するであろう。

【0078】ブロック240に示されるように、もしプログラムがユーザへ情報を表示する権限を与えられれば、この権限は、たとえば特別なウィンドウ内に、または特別なコンソール上のみに表示を許可するだけで制限されてもよい。

【0079】もしブロック238で識別された権限が選択されなければ、図11、ブロック242に示されるように、プログラムがユーザからの入力を請求する権限を有すべきか否かについてのチェックが行なわれる。もしそうであれば、この権限は、たとえば特別なウィンドウまたは端末からの請求により可能性のある制限の特定によって、資格付けされてもよい(244)。

【0080】もしブロック242で識別された権限が選択されなければ、プログラムがデジタル署名を請求する権限を有すべきか否かに関するチェックが行なわれる(246)。この点で、1組の情報を表示することによっていたずら半分のプログラムがユーザを引っかけるかもしれないが、それによって実際のデジタル署名が全く異なる組のデジタル材料へ与えられる。それゆえ、デジタル署名動作を請求する、および/または行なうP A I 権限を要求することによって、権限を付与されないプログラムは権限を付与されたプログラムの外部属性を真似るが、ユーザのデジタル署名能力を詐欺まがいの材料へ内部的に与えることを防止される。

【0081】もしプログラムがデジタル署名を請求することを許可されれば、ブロック248に示されるようにこの権限に制限が加えられてもよい。したがって、プログラムは制限された範囲、値、権限または他の特性を有する材料にデジタル署名を行なうことのみが許容されるであろう。

【0082】もしブロック246で識別された権限が選択されなければ、プログラムがロボット装置、または特

定のコンピュータ機器もしくはコンピュータ関連装置へ向ける権限を有するか否かを決定するチェックが行なわれる。もしこのような権限が選択されれば、このような機器に及ぼす制御の詳細および範囲を特定することによってこのような権限に資格が与えられてもよい(252)。

【0083】もしブロック250で識別された権限が選択されなければ、アクセスが一般にセキュリティクラスによって制限されるべきか否かを決定するチェックが行なわれる(254)。したがって、或る資源、ファイルなどが機密、機密扱いなどのような特定のセキュリティクラスと関連され得る。もしこのような権限がプログラムと関連されるべきであれば、特定のセキュリティレベルの指定を含む制限が同様に特定されてもよい(256)。

【0084】もしブロック254で識別された権限が選択されなければ、他のコンピュータ機能、または資源が制御されるべきであるか否かについてチェックが行なわれる(258)。もしそうであれば、ユーザはこのような他のコンピュータ機能、または資源についての詳細を特定することが促される(260)。

【0085】もしブロック258で識別された権限が選択されなければ、図12に示されるように、ユーザが権限の特定を終了したか否かを決定するチェックが行なわれる(262)。もしユーザが権限の特定を終了していなければ、権限選択のアレイがこの点で消耗されているために、不明の特定された権限を特定することをユーザが試みていることを示すメッセージが発行される(264)。ルーチンは図9ヘルーチン分岐して戻り、エントリポイントGでブロック210における処理を再開する。

【0086】ブロック262に定められるように、もしユーザが権限の特定を終了していれば、前に規定された権限のすべてが収集され、図2に示されるP A I 構造がデジタル署名関連のエントリを除いて完了する。

【0087】ブロック268で、P A I 構造がデジタル署名されるべきか否かを決定するチェックが行なわれる。もしそうであれば、ブロック270に示されるように、適当なデジタル署名動作がP A I 構造上で行なわれる。デジタル署名は本発明者の米国特許第4,868,877号および5,005,200号における教示に従ってか、または所望されるような従来のデジタル署名および証明技術を使用して行なわれてもよい。その後、P A I は、たとえば図3ないし図6に説明されるアプローチの1つを使用して記憶され、それによってそのプログラムと関連付けられ(272)、その後ルーチンが退出する。

【0088】図13に移って、エントリポイントFで、ブロック212ないし258に関して説明された権限の各々が選択され、選択の指示が記録された後、ルーチンは分岐してブロック274へ進み、権限の特定がディ



タル署名しているか否かを決定する。もし権限がデジタル署名されていないければ、新たに規定された権限が関連のプログラムのための権限情報へ加えられ(280)、ルーチンは図9のエントリポイントGでブロック210へ分岐して戻る。

【0089】もし権限がデジタル署名されるべきであれば、向上した証明(権限を有する)が本発明者の米国特許第4,868,877号および5,005,200号に従って使用されているか否かについてのチェックが行なわれる(276)。もしそうでなければ、ルーチンは前述のようなブ

ロック280へ分岐する。

【0090】もし向上したデジタル署名が使用されていれば、前述の特許に説明されるようなユーザの向上した権限証明がこの特定のプログラムの権限特定を割当ててを許可するか否かを決定するチェックが行なわれる。もし向上した権限証明がこのような権限の割当てを許可すれば、ブロック280の前述の処理が行なわれる。もしそうでなければ、ブロック282に示されるように、「あなたの証明はこのレベルのプログラム権限の割当てを許可されていません。」というメッセージがユーザへ発行される。ルーチンは図9およびブロック210の処理のためのエントリポイントGへ分岐して戻る。

【0091】図14および図15は、そのプログラム権限情報に従って実行されているプログラムの処理を制御するためのスーパーバイザプログラムの動作のシーケンスを示す。図14に示されるプログラム「X」およびそのプログラム権限情報の処理は、コンピュータがスーパーバイザルーチンを実行している間に開始される。図14の300で示されるように、呼出プログラムは実行のためにプログラムXを呼出す。その後、プログラム制御ブロックがプログラムXのために作成される。作成されたプログラム制御ブロックは、プログラムが呼出されることを許可されることが決定され、かつ確認がうまく完了されるまで、実行スタックの先頭に加えられないであろう。したがって、もしプログラムがセキュリティチェックを失敗すると、それはプログラム実行鎖に置かれまいであろう。「試験的」プログラム制御ブロックを作成することに加えて、呼出されたプログラムはブロック302の処理中に適当なプログラムディレクトリを介して位置決めされるであろう。

【0092】その後、ブロック304で、PAIが前述のいわゆる「セーフティボックス」内にプログラムXを置くようにプログラムXとまだ関連されているか否かを決定するチェックが行なわれる。このPAIは前述のようなその特定の応用によって署名されても、されなくてもよい。

【0093】もしいかなるPAIもプログラムとまだ関連されていないければ、プログラムが製造者からの関連の署名された「系統(pedigree)」を有するか否かを決定するチェックが行なわれる(306)。したがって、も

し周知のプログラム製造者がパブリックキー、またはデジタル証明でプログラムに署名しているならば、所望ならこのようなプログラムが、製造者がどれだけ信用しているかによって所望のどのような権限レベルを割当てられてもよく、システムはこのようなプログラムの実行を許可してもよい。製造者からのこのようなデジタル署名は、プログラムが製造者によって生成されたときと全く同じであるか否かを決定されることができるので、関連のプログラムがウィルスによって感染されていないということを検証するために使用され得る。

【0094】もしブロック306のチェックが、ブロック308で製造者からのデジタル署名があるということを示せば、ユーザによって前に確立されている、(かつユーザが前に信用を確立した製造者によって署名されている)信用基準があるとすれば、デジタル署名を確証し、適切である証明および権限チェックのいかなるものも行なうことによって、製造者の「系統」が確認されるであろう。権限を委任するデジタル署名を行なうための機構は、本発明者の米国特許第4,868,877号および5,005,200号に詳細に特定され、これらの特許は引用によってここに明確に援用されている。

【0095】ブロック308の確認動作の結果によって、ブロック320で、製造者の系統が認容可能であるか否かについての決定が行なわれる。もし製造者の系統が認容可能でなければ、ルーチンはブロック324へ分岐し、そこでプログラムの実行が以下に説明されるであろうように抑制される。

【0096】もし製造者の系統が認容可能であれば、ルーチンはブロック326へ分岐し、そこで記憶域がプログラムについて割当てられ、プログラムは以下に詳細に説明される態様でロードされる。

【0097】ブロック310で示されるように、もしPAIがプログラムXに関連していると決定されれば、PAIが署名されるか否かを決定するチェックが行なわれる。もしPAIが署名されれば、ブロック316で示されるように、署名が確認される。現在好ましい実施例において、署名が証明階層を通して確認される。署名が有効であるか、それらが呼出人によって信用されているか、およびプログラムによって委任された権限が署名者によって委任されていることを許可されるかを決定するための好ましい方法論が本発明者の米国特許第4,868,877号および5,005,200号に教示される。これらの特許に示されるように、信用レベルはどの高レベルのパブリックキー、および/または超証明子(metacertifier)がユーザによって信用されるように特定されているかによって決定されてもよい。代替的により慣習的なデジタル署名技術が使用されてもよい。

【0098】ブロック316の処理によって、ブロック322で署名が有効であり、権限が付与され、かつ信用されているか否かの決定が行なわれる。もし署名が有効

であると決定されなければ、ルーチンはブロック324へ分岐し、そこでプログラムXの実行が抑制される。

【0099】もしブロック310のチェックが、PAIが署名されていないということを表わせば、312で特定のシステム、またはアプリケーションが、PAIが署名されることを要求するか否かについてのさらなるチェックが決定される(312)。もし、たとえばユーザ生成されたプログラムがユーザ自身の使用のために実行されていれば、プログラムは分配されておらず、かつユーザは自分が行なったことを信用しているので、署名は必要でないであろう。もしブロック312でデジタル署名が必要でなかったと決定されれば、ブロック318は署名されないPAIを受取り、かつこれを使用し、記憶域が割当てられ、プログラムXがロードされるであろう(326)。

【0100】もしブロック312でデジタル署名が必要であると決定されれば、ブロック314で、明示PAIまたは署名されないPAIを有さないプログラムのための「最小の」権限省略値をシステムが有するか否かについてのチェックが行なわれる。したがって、たとえばシステムは、いかなる永続ファイルも変更することを試みない限り、最小の権限省略値下でプログラムが実行することを許可してもよい。もし最小の権限省略値がなければ、プログラムの実行は抑制される(324)。プログラム実行の抑制過程において、エラーコード、またはメッセージが呼出プログラムへ戻されるであろう。たとえば、「プログラムXは有効な署名された権限を有さない」というメッセージが呼出プログラムへ表示されてもよい。ルーチンはブロック410へ分岐し、そのブロックはさらに後に説明されるであろうように実行を実際に抑制するように作動する。

【0101】もしブロック322および316の処理が署名が有効であることを明らかにすれば、ブロック326の処理が行なわれる。まず、記憶域がプログラムについて割当てられる。プログラムはメモリ内にロードされてもされなくてもよく、このメモリはスーパーバイザのみがコンピュータシステムおよびプログラムの性質に組込まれた制約によって変えることを許容される。もしプログラムがそれ自体を変更すれば、それはスーパーバイザのみが変えることを許容されるメモリ内にロードされることができない。

【0102】その後、プログラムXのプログラム権限付与情報が、もしあるならば呼出プログラムのPCBと関連のPAIと適当に組合わされる。多数のPAIのものを組み得る組合わされたPAIはプログラムによって一般に変更され得ない記憶域内に記憶され、PAIのアドレスは図8のフィールド156に示されるようにプロセス制御ブロック(PCB)内に記憶される。したがって、プログラムXはもし呼出プログラムによって呼出されれば、それ自身の制約のすべてを受けるとともに、呼

出プログラムの制約とある点で組合わされており、その集合制約はプログラムXのPAIに具体化される。この態様において、呼出プログラムは単に別のプログラムを呼出すことによってその割当てられた境界を超えることが許可されない。プログラムのPAIが、プログラム自体の固有の性質、および現在の環境に適用可能である方策によって、それを呼出すプログラムのPAIと組合わされ得る、多くの代替方法がある。組合わせ方法でさえそれ自体、呼出す、または呼出されたプログラムのいずれか、または両方のPAI権限、または資格付けるものの1つであり得る。

【0103】たとえば、呼出されたプログラムをその「通常の」PAI権限の小さい方およびその呼出プログラムのそれに制限し、呼出プログラムが呼出されたプログラムの大きい方の権限をいわずに半分に誤用してそれ自身の制限を巧みに回避できないことを保証することが穏当である。

【0104】他方、それら自身の処置を注意深く確認する呼出されたプログラムでは、呼出されたプログラムにそれを呼出すプログラムより大きい固有の権限を許容することが可能であり得、このように高感度の資源がより広い使用を、信用されたサブプログラムを通してこのような使用を取り次ぐことによって利用可能にされ得る。このような組合わせの可能性は基本的制御システムの設計者によってだけではなく各プログラムへの権限の割当てを行なう者によっても注意深く検討されねばならない。その後、プログラムはロードされ、プログラムのハッシュがプログラムのPAIに特定されるアルゴリズムに基づき計算される。プログラム314に戻って、もしシステムが最小の権限省略値を有すると決定されれば、ブロック328に示されるように、最小の省略値権限が使用される。このような最小の省略値権限はもしあれば呼出プログラムのPAIと適当に組合わされ、ブロック326に関連して上に説明されたように新しいPCB内に挿入される。PAIの記憶域はプログラムが一般に変えることができないメモリから割当てられる。その後、記憶域がブロック326に関連して上に説明されたようにプログラムについて割当てられ、アドレスがPCB内に保管される。省略値権限を使用するブロック328の処理はプログラムのハッシュの計算を含まず、ルーチンはブロック334へ分岐して、プログラム実行に備える。

【0105】ブロック330に移って、ブロック326の計算されたプログラムハッシュがPAI内に記憶されるハッシュと一致するか否かを決定するチェックが行なわれる。もしハッシュが一致しなければ、ルーチンはブロック332へ分岐し、そこで「プログラムXが変えられた、または損害を受けた。」というようなエラーメッセージが呼出プログラムへ送られ、ルーチンはブロック410へ分岐し、プログラムの実行を抑制する。

【0106】ブロック334において、ハッシュが一致すると決定された後、またはブロック328における処理の後、プログラムが初期実行のために用意される。初期実行のための用意は、初期状態のセット、およびプログラムのPCBにおける情報の「再開」を含み、それによってプログラムは適切なエントリポイントで開始するであろう。さらに、プログラムのPCBは実行スタックの先頭に置かれるであろう。

【0107】図15へ移り、ブロック336において、実行スタックの先頭に置かれると、現在のプログラムは実行を開始するか、または再開する。ブロック330ないし410で起きる処理は、プログラムを実行するために慣用的に行なわれる動作を含む。処理動作がこれよりPAI処理を含むそれらの動作に重点をおいて説明されるであろう。ブロック336において、スーパーバイザは保管された「再開」点で、様々なレジスタの状態を再ロード（またはロード）して、プログラムが最後に中断された（または初期化された）時までその点でそれらの状態を示すことによって、プログラムを継続する用意をする。さらに、たとえばスタックポインタなどのようなシステム状態情報が使用されている特定のシステム環境によって回復される。

【0108】ブロック336の処理の後、もしアプリケーションプログラムが実行されていると、システムは「スーパーバイザ」モードから「隔離 (isolation)」モードへ切換え、それによってプログラムは隔離モードで実行を再開する（338）。隔離モードにおいて、プログラムは「スーパーバイザ」モードへコンピュータを切換え戻す保護されたスーパーバイザ呼出を介して以外はコンピュータ資源に影響を及ぼすことができない（或る場合、および或る環境において、プログラムが「スーパーバイザ」モードで実行するように設計され、要求されることが可能であろうことが注目される。この場合、プログラムがそのPAIで規定されるように適切に権限付与されると仮定すれば、或る点において「スーパーバイザ」機能を使用し、「スーパーバイザ」状態動作が可能のようにそのPCBの状態をセットするであろう。このような場合、その状態をチェックし、もしセットしてあれば、プログラムスーパーバイザ状態へ制御を与えることが適当であろう。）。

【0109】ブロック340において、プログラムが制御された「スーパーバイザ」機能を要求していることが推定される。このような環境下で、たとえば予め定められた状態ワードをセットすることによってコンピュータを「スーパーバイザ」モードへ切換え、保護されたシステム監視中断ルーチンへ制御を送る。プログラムの再開位置はプログラムのPCBに保管され、他の適切なシステム状態はPCBに保管される。その後、アクセスされるべき機能および資源が決定され、アクセスの性質は、たとえば読出、変更、および削除などである。

【0110】さらに、ブロック340において、プロセス制御ブロックに記憶されるPAI情報から調査が行なわれる。ブロック340の処理を補うため、またはそれに関連して、ブロック342において調査されたPAIが要求された資源へのアクセスを許容されたか、または要求された機能を行なうことを許容されたかを決定するチェックが行なわれる。たとえば、もし電子郵便を使用することが試みられるならば、プログラムが電子郵便機能を行なう権限を付与されたか否か、もしそうであれば、郵便が1組の郵便識別子に制限されるか否かを決定するチェックがPAIから行なわれる。

【0111】もし342のチェックで、PAIが試みられた機能、または資源のアクセスを許容しないということを示すメッセージが生成され、資源または機能へのアクセスが否定され、適当なエラーコード、またはメッセージが生成される。ブロック350において、アクセスの達成を試みているプログラムが、アクセスを否定されているということを知らされるべきか否かを決定するチェックが行なわれる（350）。もしブロック350のチェックで、プログラムがそう知らされるべきであると明示されれば、ブロック352においてプログラムがその要求がうまく行なわれず、かつ抑制されるようにせしめたアクセス侵害の型を示すメッセージとともに実行を再開することが許容される。ルーチンはプログラムの実行を再開するためにブロック336へ分岐して戻る。このような環境下でプログラムは、たとえばそのPAIが特定のファイルについて権限を讀出す権限のみが与えられているということを知らされてもよく、一方でそのファイルへ書込む試みが行なわれた。もしブロック350のチェックが、呼出プログラムが知らされるべきでないと示せば、適当な状態および関連のメッセージ（呼出プログラムのための）が生成され、特定されないアクセス侵害による終了を示す356。

【0112】もしブロック342のチェックが、PAIが機能または資源へのアクセスを許容すると明示すれば、ブロック346で、慣用的なアクセス制御を与え、プログラムのユーザがまだ自分の権限の境界内にあることを確実にするためのチェックが行なわれる。このチェックは、機能または資源の要求がこの特定のユーザのためのシステムによって許容されるものの範囲内であることを確実にする。したがって、PAIはプログラムが或るクラスのファイルにアクセスすることを許容するであろうが、特定のユーザに関するセキュリティレベルがそのようなファイルへのそのユーザのアクセスを許容しないかもしれない。ブロック346は従来のセキュリティ技術を適用して権限を適切に付与されないユーザからシステムを保護する。このチェックはたとえば、署名すると最初にシステム内に入るユーザ識別コードに基づいて

もよい。ブロック348で示されるように、もしユーザが権限付与されなかったならば、プログラムがユーザのアクセス能力を侵害することを試みているので、アクセスが否定され、適切なエラーコード/メッセージが生成される。その後、ブロック350および352に関して上に説明された処理が開始される。

【0113】もしユーザがブロック346の処理によって定められるように権限付与されれば、ブロック354において示されるように機能が行なわれる。もし機能がプログラムの退出であれば、ルーチンはブロック358を介してプログラムから退出し、そこで関連のPCBが実行スタックの先頭から取除かれる。

【0114】さらなる終了処理がブロック410で行なわれ、そこでプログラムの実行中に蓄積されたいかなる漂遊記憶および資源も解放され、それは必要に応じてプログラムXに割当てられた記憶およびそのPAI記憶を含む。PCBおよびすべての関連の記憶が解放され、エラーコードおよびメッセージを適切に含む最終状態情報がプログラムの呼出人に提示される。その後、ルーチンはブロック336へ分岐して戻り呼出プログラムを再開する。

【0115】最後に、ブロック354へ戻って、もしプログラムの呼出が行なわれるべき機能であれば、ルーチンは図14のブロック300へ分岐し、適切なプログラムを呼出す。

【0116】この発明は現在最も实际的、かつ好ましい実施例であると考えられるものに関して説明されてきたが、この発明は開示される実施例に限定されるべきではないと理解されるべきであり、前掲の請求項の精神および範囲内に含まれる様々な変更および同等の配置を包括することが意図される。

#### 【図面の簡単な説明】

【図1】この発明と関連して使用され得る例証的なコミュニケーションシステムを示すブロック図形式の図である。

【図2】プログラム権限情報データ構造の例示の図である。

【図3】プログラム権限情報をプログラムに関連づける

例証的な方法を例示する図である。

【図4】プログラム権限情報をプログラムに関連づける例証的な方法を例示する図である。

【図5】プログラム権限情報をプログラムに関連づける例証的な方法を例示する図である。

【図6】プログラム権限情報をプログラムに関連づける例証的な方法を例示する図である。

【図7】不明の出所のプログラムと関連して、ユーザがこの発明をどのように使用し得るかを例示する一般的なフローチャートの図である。

【図8】この発明の例証的な実施例に従うプログラムコントロールブロックデータ構造の例示の図である。

【図9】プログラム権限情報を確立するためのプログラムの動作のシーケンスを正確に叙述するフローチャートの図である。

【図10】プログラム権限情報を確立するためのプログラムの動作のシーケンスを正確に叙述するフローチャートの図である。

【図11】プログラム権限情報を確立するためのプログラムの動作のシーケンスを正確に叙述するフローチャートの図である。

【図12】プログラム権限情報を確立するためのプログラムの動作のシーケンスを正確に叙述するフローチャートの図である。

【図13】プログラム権限情報を確立するためのプログラムの動作のシーケンスを正確に叙述するフローチャートの図である。

【図14】プログラム権限情報を処理する際のスーパーバイザプログラムによって実行される動作のシーケンスを例示する図である。

【図15】プログラム権限情報を処理する際のスーパーバイザプログラムによって実行される動作のシーケンスを例示する図である。

#### 【符号の説明】

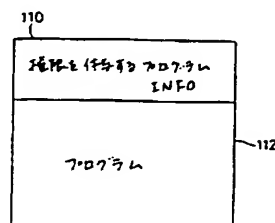
2 プロセッサ

4 ディスプレイ

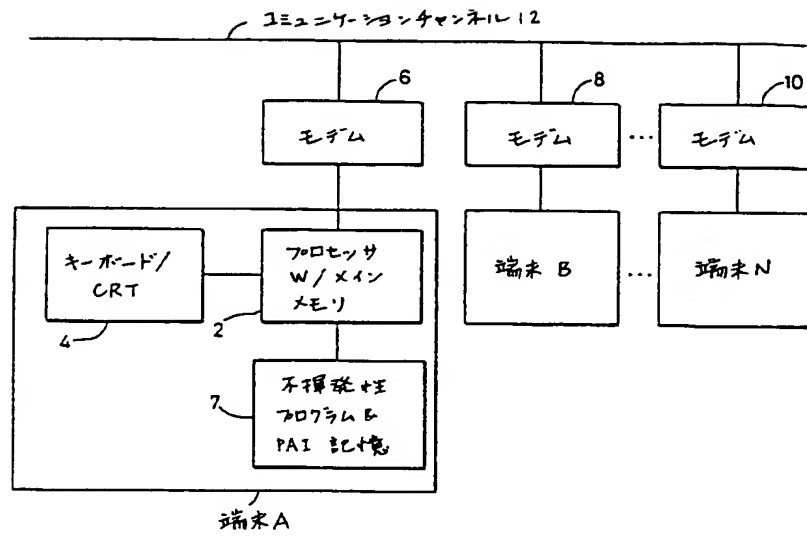
7 メモリ

12 コミュニケーションチャンネル

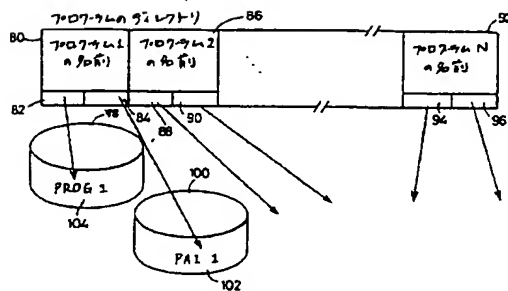
【図4】



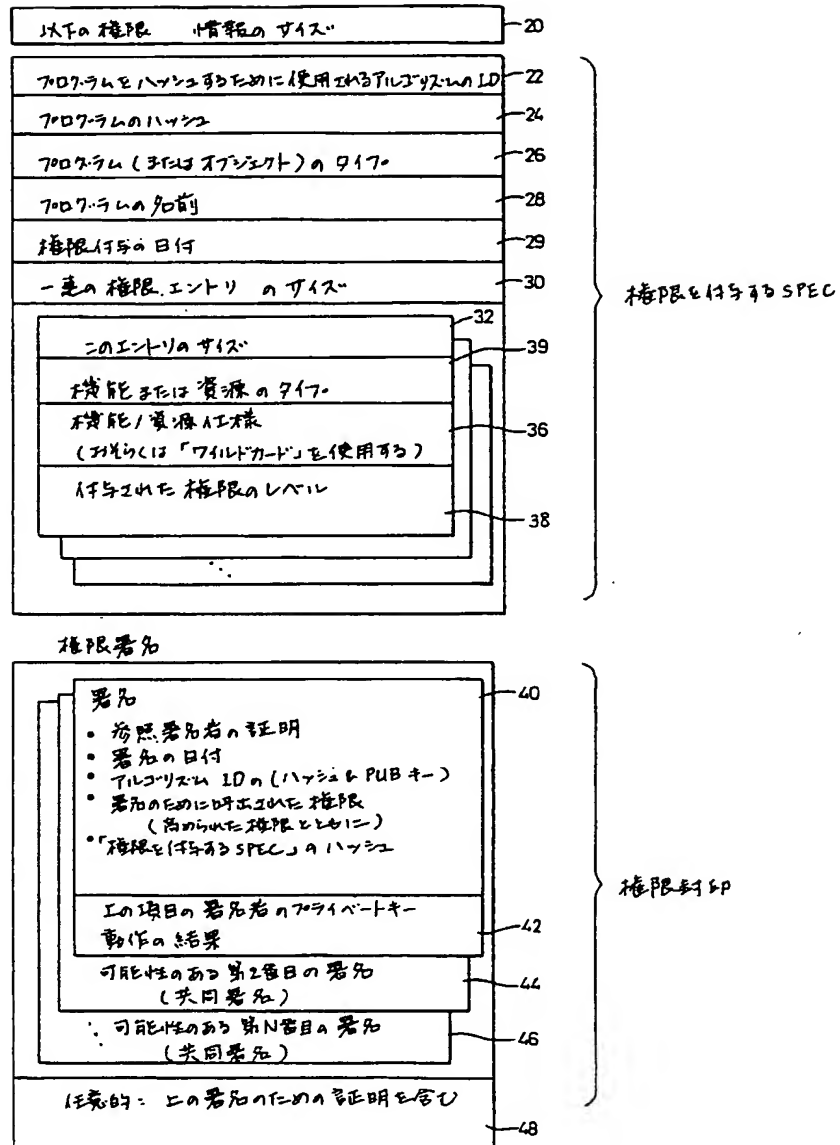
【図1】



【図3】

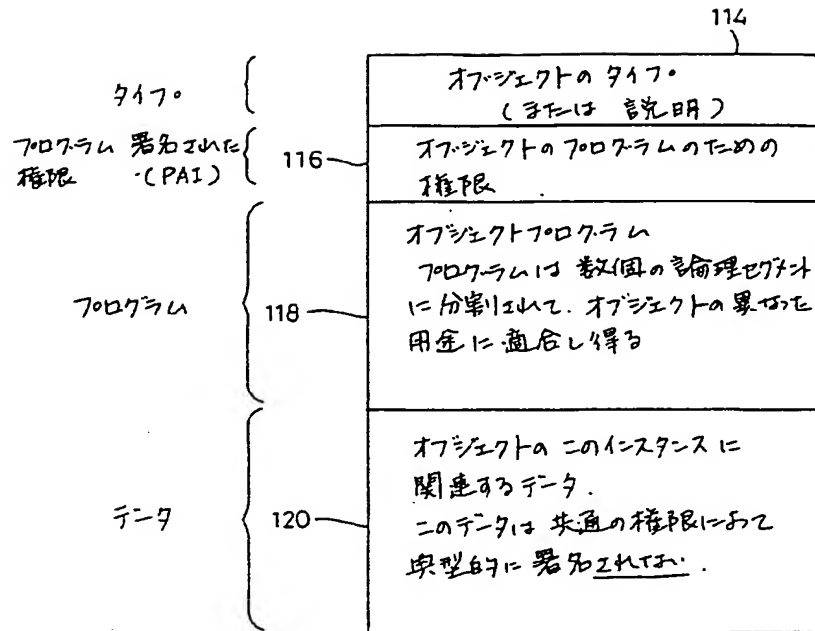


【図2】

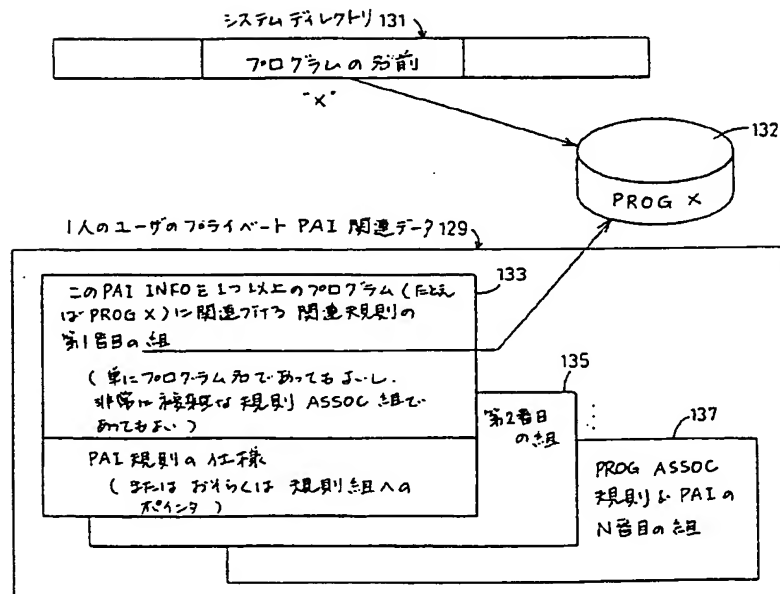




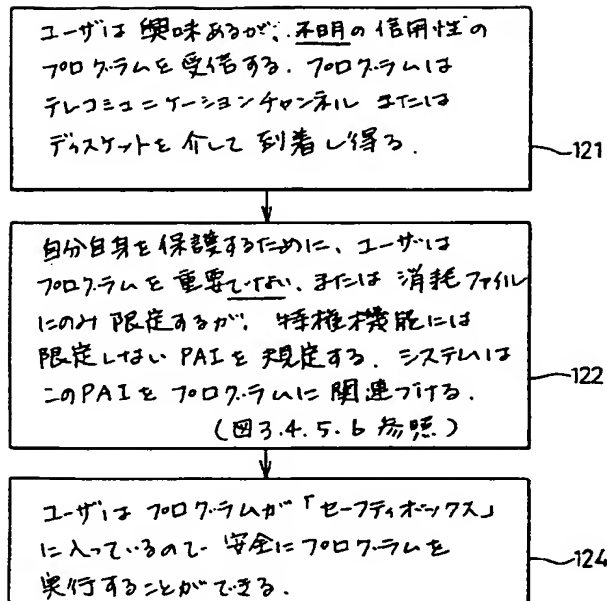
【図5】



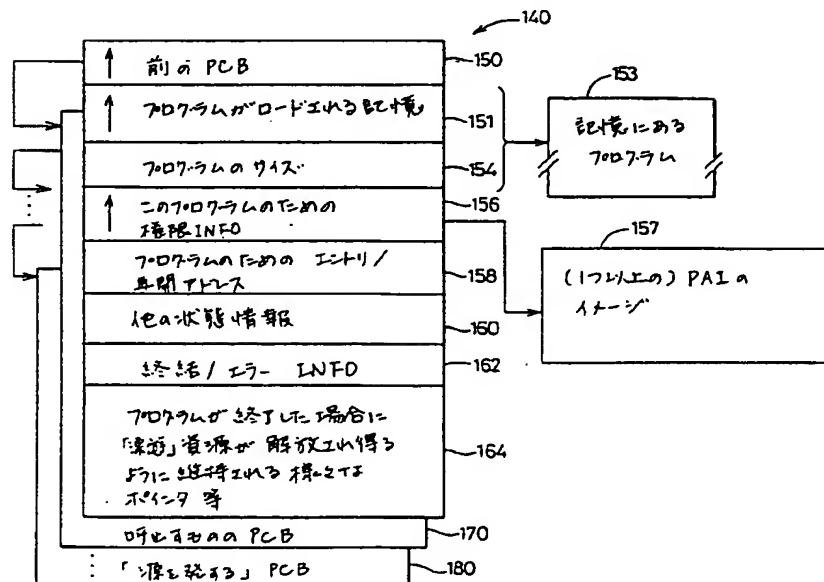
【図6】



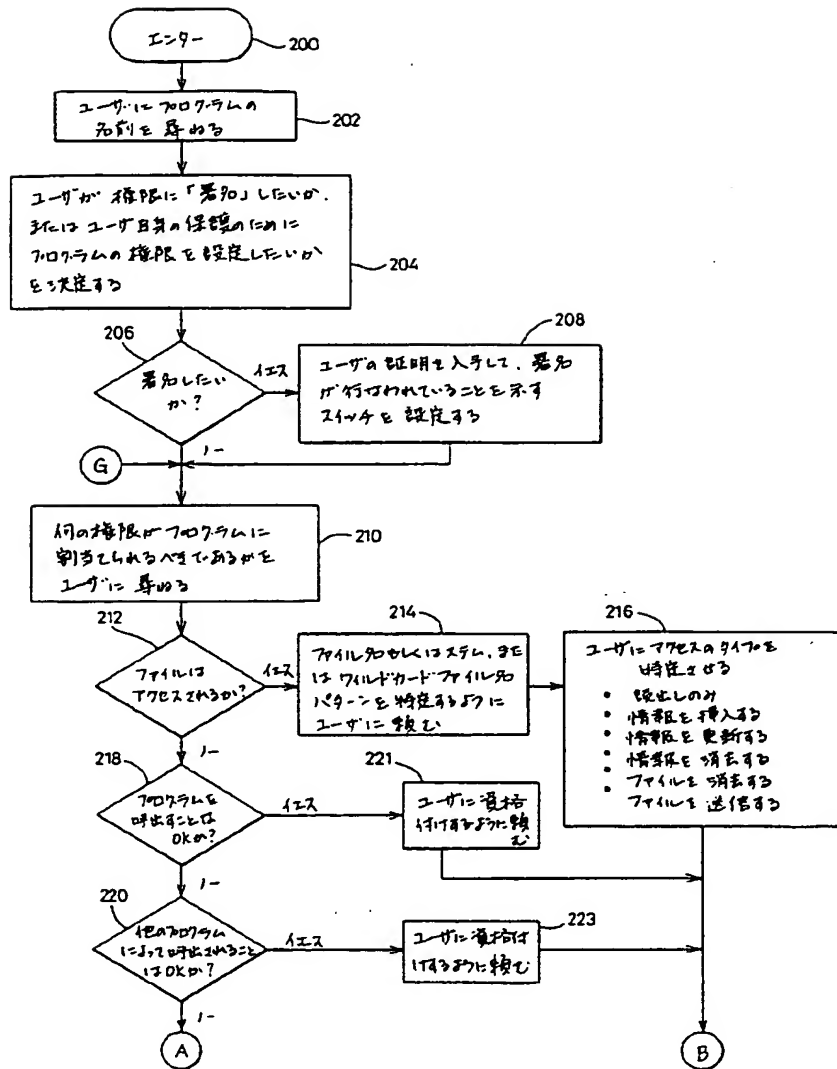
【図7】



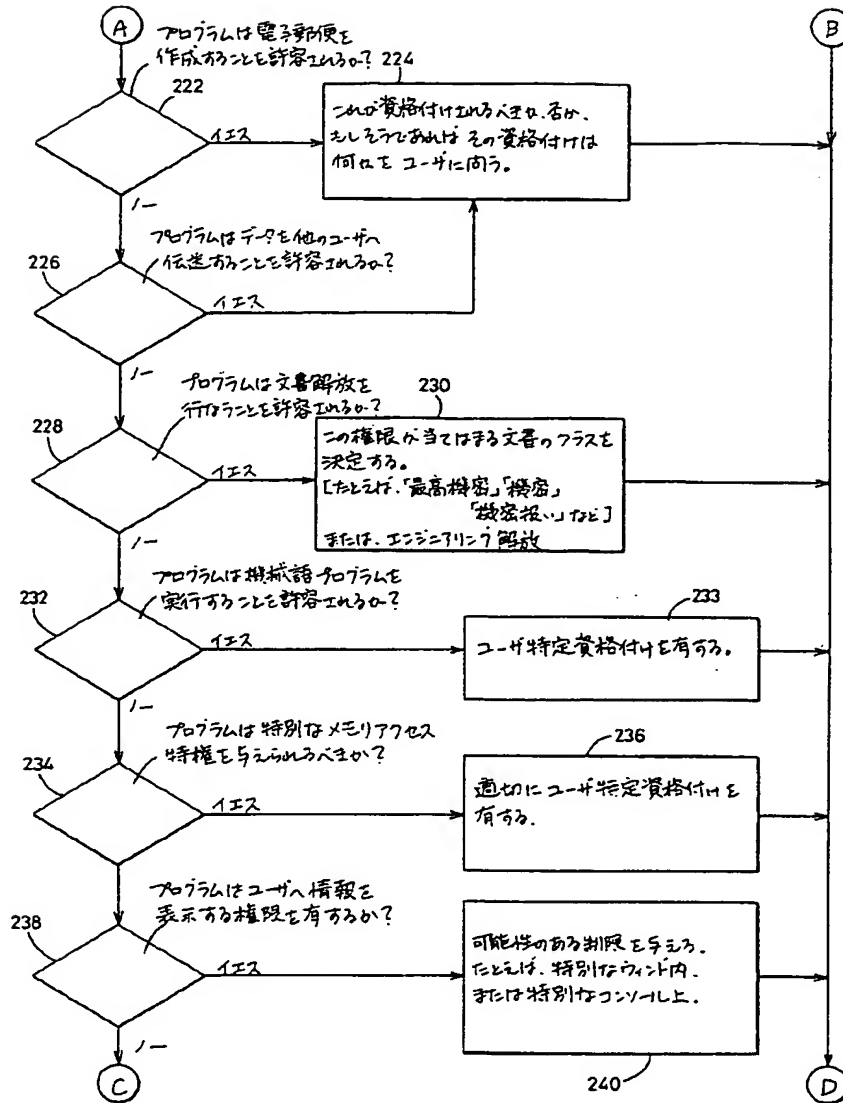
【図8】



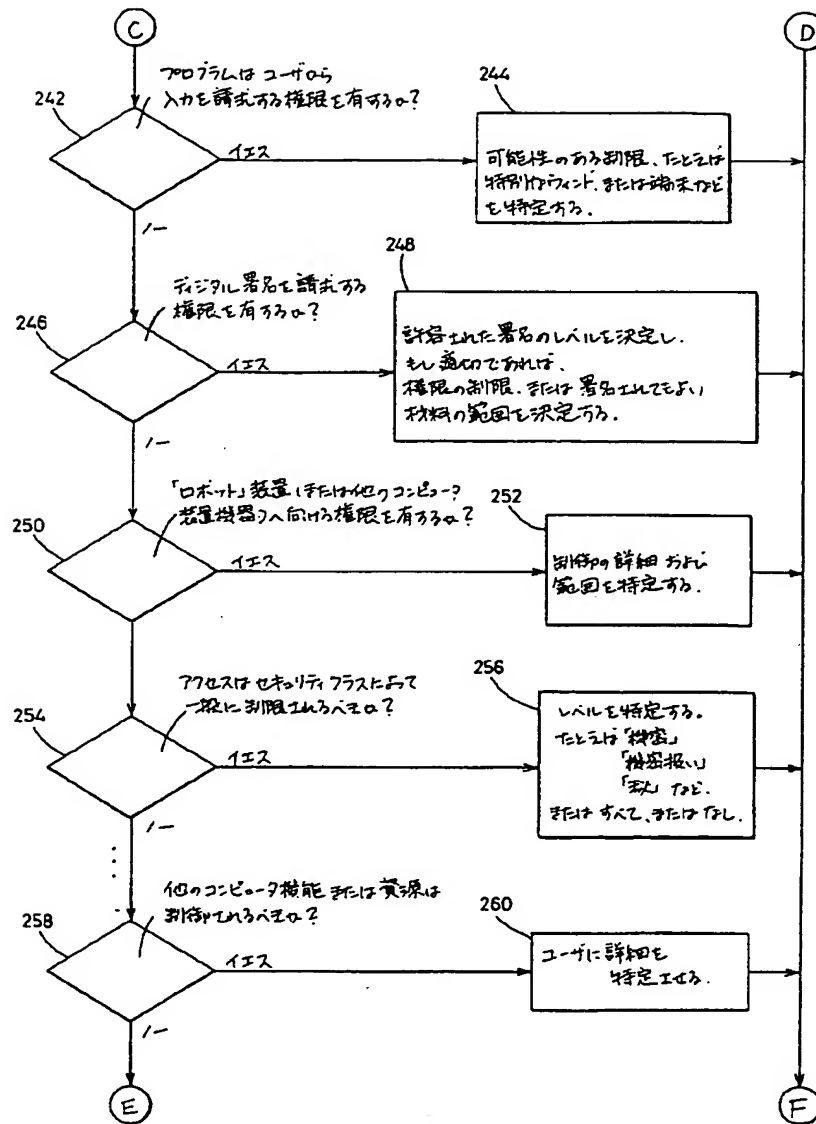
【図9】



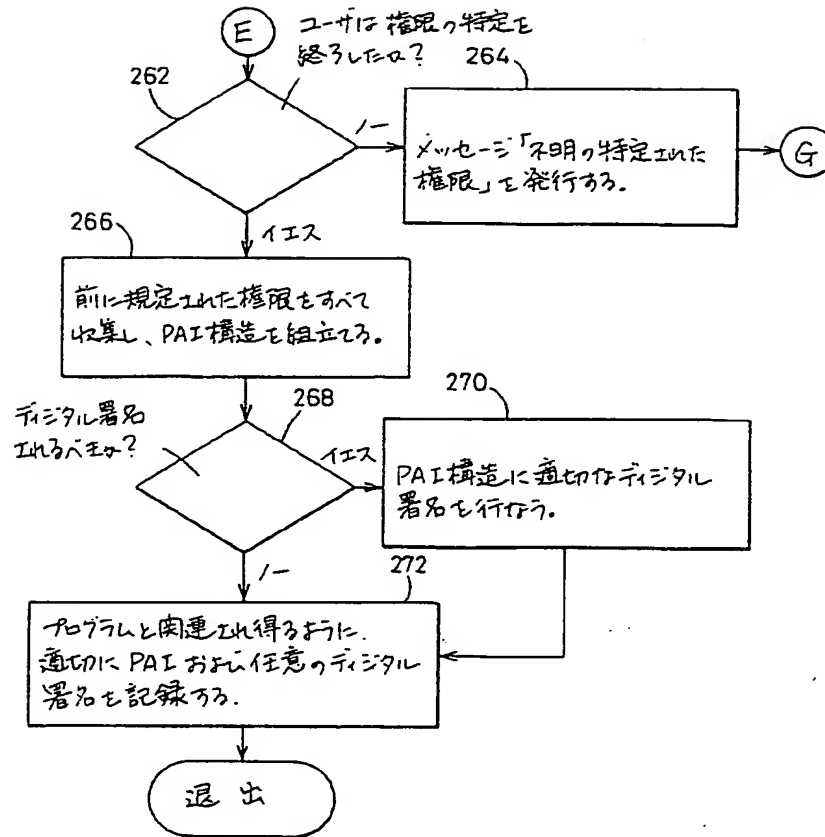
【図10】



【図11】

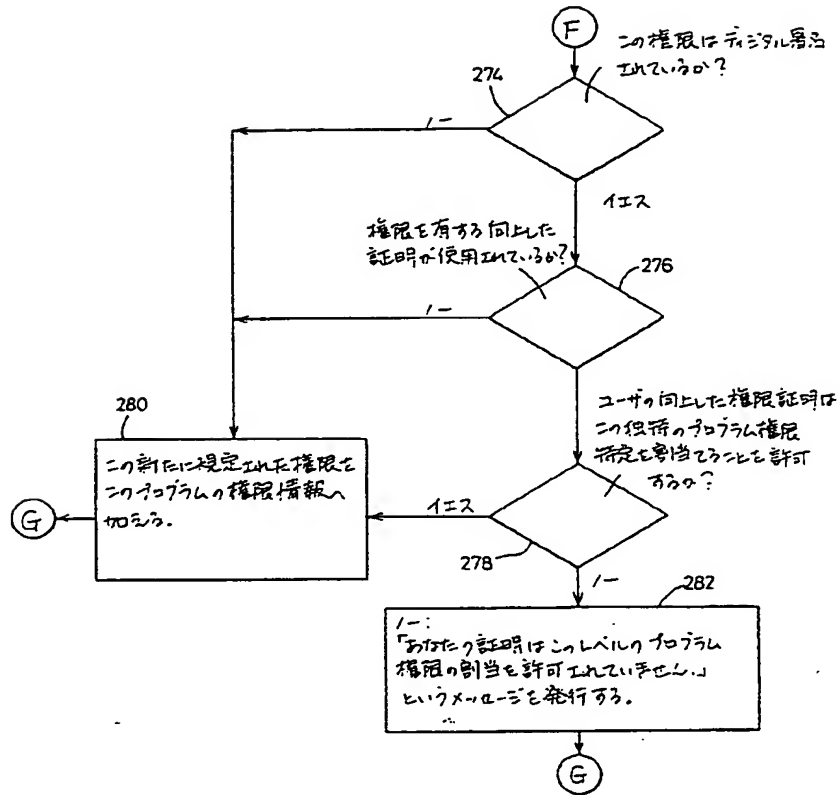


【図12】

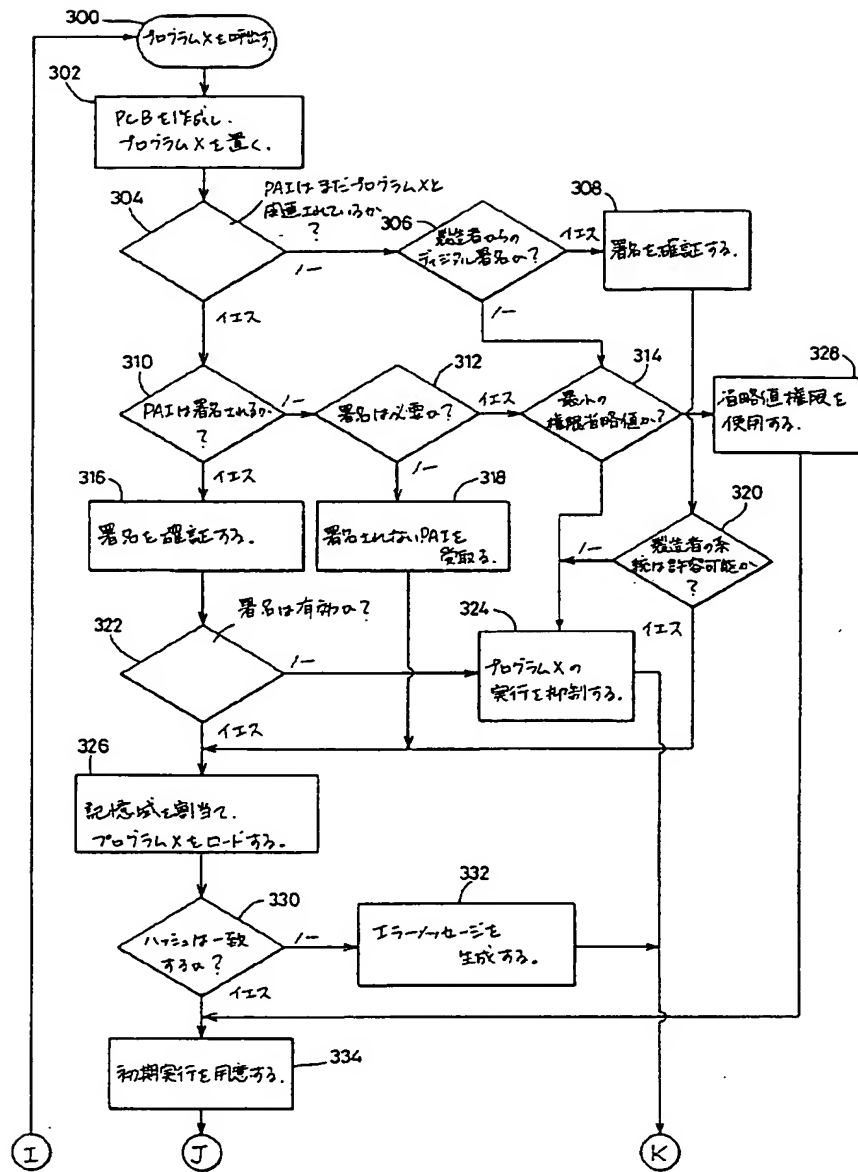




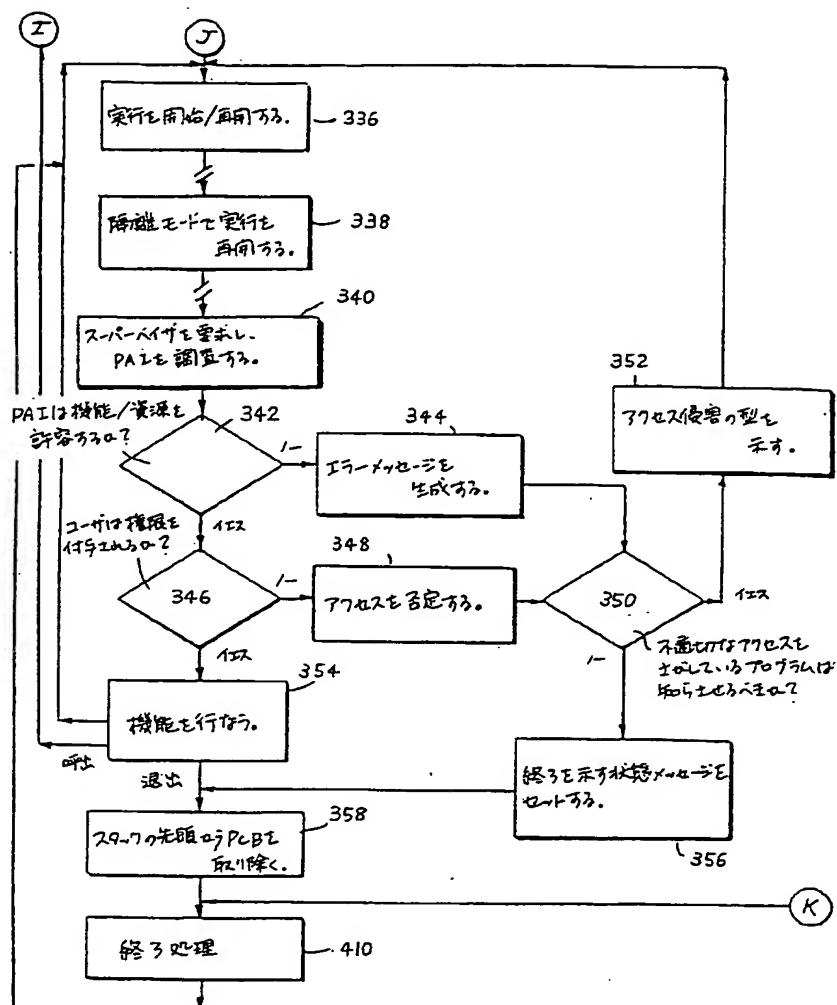
【図13】



【図14】



【図15】



フロントページの続き

(51)Int.Cl.5  
H04L 9/12

識別記号

庁内整理番号

FI

技術表示箇所